

DOCUMENT RESUME

ED 168 575

IR 007 116

TITLE Sixth New Zealand Computer Conference (Auckland 78) .
Volume II, The Proceedings.

INSTITUTION New Zealand Computer Society, Auckland.

PUB DATE 78

NOTE 158p.; Not available in hard copy due to small type size of original document; For related document, see IR J07 115

AVAILABLE FROM New Zealand Computer Society, National Executive Office, P.O. Box 2123, Wellington (Vol. I \$6.00 and Vol. II \$4.00)

EDRS PRICE MF01 Plus Postage. PC Not Available from EDRS.

DESCRIPTORS Communication (Thought Transfer); Computer Assisted Instruction; Computer Oriented Programs; *Computers; *Computer Science; Dial Access Information Systems; Electronic Data Processing; Information Networks; Information Processing; *Information Systems; Programing Languages; *Technological Advancement; Telecommunication

ABSTRACT

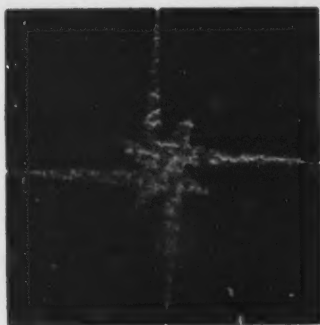
Speeches made by five keynote speakers, summaries of five forum sessions, and one paper are included in this collection of presentations at a conference on the application of computer technology in New Zealand. The keynote addresses include "Analysis of Key Factors for Motivation of Data Processing Professionals," by J. Daniel Conger; "Communication is a Two Way Process," by Dr. Brian Cox; "Computers and Communications," by Sandy Douglas; "What Is To Be Done?" by Sidney Michaelson; and "New Directions in Data Processing," by H. Wayne Nelson. Forum sessions discussed education, plans for the next conference, user aspects of terminal design, selection of a computer manager, and telecommunications. The additional paper, by Owen Hanson, discusses "Choosing Between Sequentially and Indexed-Sequentially Organised Files for Sequential Processing." (JEG)

* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *

ED168575

U.S. DEPARTMENT OF HEALTH,
EDUCATION & WELFARE
NATIONAL INSTITUTE OF
EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY.



SIXTH
NEW ZEALAND
COMPUTER
CONFERENCE
AUCKLAND 78

"PERMISSION TO REPRODUCE THIS
MATERIAL IN MICROFICHE ONLY
HAS BEEN GRANTED BY

C.C. Bradley

TO THE EDUCATIONAL RESOURCES
INFORMATION CENTER (ERIC) AND
USERS OF THE ERIC SYSTEM."

2

THE PROCEEDINGS VOLUME II

IR007116

CONTENTS

PAGE NO.

KEYNOTE SPEAKERS

PROFESSOR J. DANIEL COUGER

- "Analysis Of Key Factors For Motivation
Of Data Processing Professionals" 1

DR BRIAN COX

- "Erua Nga Waahi O Te Pa" 20

PROFESSOR A.S. (SANDY) DOUGLAS

- "Computers and Communications" 40

PROFESSOR SIDNEY MICHAELSON

- "What Is To Be Done?" 60

MR H. WAYNE NELSON

- "New Directions in Data Processing" 81

FORUM SESSIONS

Education 118

Next Conference 125

User Aspects of Terminal Design 127

Selection of a Computer Manager 129

Telecommunications 134

PAPER

OWEN HANSON

- "Choosing Between Sequentially and Indexed -
Sequentially Organised Files for Sequential
Processing" 136

KEYNOTE ADDRESS

"ANALYSIS OF KEY FACTORS FOR MOTIVATION
OF DATA PROCESSING PROFESSIONALS"

by

J. DANIEL COUGER

Elected U.S. Computer Science Man of The Year (1977), in recognition for his outstanding contributions to the development of Computer education for management, Professor Couger has achieved significant distinction in the U.S.A.

His 1966 survey of computer curriculum in accredited U.S. schools of business revealed that only 11% required computer proficiency of graduates. This led to his development of a computer science curriculum to overcome this deficiency, and a 1977 survey revealed that 94% of U.S. accredited schools now require computer proficiency to graduate.

In addition to is curriculum work, Professor Couger has published extensively in the area of information systems design. He has served as an officer in three international professional societies and has lectured in 33 countries on six continents.

Consulting activities undertaken by Professor Couger on computer science and information systems curriculum are widely applied to U.S. universities and schools of management. He also lectures on advanced business systems analysis techniques, and for data processing management, lectures on key factors for the motivation of data processing personnel.

Professor Couger's attendance at our Conference is courtesy of I.B.M.

Introduction

Now more than 25 years old, the computer profession may be assuming some of the behavioral problems characteristic of older disciplines, such as engineering and accounting.

According to Work in America, worker alienation is one of the major problems in organizations today. The report indicates that the job itself has not kept pace with the changes in our society such as worker attitudes, aspirations, and values.

Managers have attempted to reduce worker alienation and improve the quality of work life by trying a plethora of ideas for the humanization of work. Some past and present ideas are: sensitivity training, management-by-objectives, shortening of the workweek, greater worker involvement in the decision-making process, etc.

Signs of job dissatisfaction are also emerging in the computer field. Although the aura of excitement of a dynamic field continues to attract persons into the profession, it no longer retains them. Individuals are seeking a more meaningful experience than just being a part of a fast growing profession. The job itself must produce the essential elements of satisfaction - the peripheral benefits of being at the forefront of technology are important, but alone are insufficient to keep a person motivated.

The behavioralists have identified factors key to motivation and satisfaction in other disciplines. But are they the same for our profession? Even if they are identical, does each factor have the same degree of importance?

We began our research project with the hypothesis that data processing professionals are different. We sought to identify key variables in job satisfaction and motivation. The ultimate objective is to identify ways in which jobs can be redesigned to improve satisfaction and productivity.

The purpose of this paper is to present our tentative findings on the motivating potential of jobs within our career field. Additionally, I will report tentative findings about the motivation patterns of the people in our profession.

Organizing the Research.

Our research team (a computer scientist and a behavioral scientist) organized the project into the following phases:

Phase I (1977-1978): Identifying Norms for the Field

Data processing personnel were surveyed in 25 organizations, both in government and industry. Included were seven industries and three levels of government (city, state and federal).

Phase II (1978-1979): Interindustry Comparison

Five to eight firms within five different industries will be surveyed, to determine interindustry differences. The software development industry will be one of those surveyed. Government versus private industry comparisons will also be made.

Phase III (1978-1980): Job Redesign

Utilizing techniques proven in other industries, data processing jobs will be analyzed for ways to improve job satisfaction and employee motivation and productivity. This research will include jobs within:
a data processing unit: key entry, data control, computer operators, programmers, programmer/analysts, analysts, supervisors and managers. These job redesign projects will be undertaken in private industry as funds become available.

Profile of the Organizations Surveyed

The non-government organizations surveyed, represented the following industries: food processing, airlines, electronics, retailing, banking, insurance, mail order. Their data

processing organizations ranged in size from 25 to 150 employees. They represented all geographic regions of the United States.

The governmental organizations were located in 15 states. Their data processing organizations varied in size from 30 to 200 employees.

Over 500 d.p. professionals (analysts, programmer/analysts, and programmers) were surveyed. More than 1,000 personnel in other d.p. jobs were surveyed.

Scope of Survey for this Paper

This paper analyzed only the results of the survey of data processing professionals: analysts, programmer/analysts, and programmers. Since data gathering and processing continued up until May 15, it was not possible to complete the paper on time for publication in the Proceedings of the National Computer Conference.

This paper will be published in a summer issue of Datamation. Survey results for all other d.p. jobs, including supervision, will be reported in two other papers to be published in later issues of Datamation.

Selection of the Survey Instrument

We originally sought to design a survey instrument peculiar to the computer field. After two months of evaluating various instruments we decided to utilize instead the Job Diagnostic Survey (JDS) developed by Greg R. Oldham (University of Illinois) and J. Richard Hackman (Yale University) for two principal reasons:

1. The Hackman/Oldham instrument is conceptually sound. Its validity and reliability have been substantiated* in studies of more than 6,000 subjects on more than 500 different jobs in more than 50 different organizations.*

2. A major objective is to compare our results with prior studies of personnel in other professions. Our hypothesis on the difference between d.p. professionals and other personnel could be tested.

We expanded the survey questionnaire to include other elements: employee perceptions on relative importance of problems relating to maintenance, realistic work schedules, access to the computer, access to supervisors and access to others (e.g., users or personnel in other departments whose work affected ours). Also added to the survey instrument was a section on the relative importance of eight categories of compensation.

All survey results were analyzed via the computer packages for Analysis of Variance (ANOVA) and SPSS (Statistical Processing for the Social Sciences).

- * As with every instrument in use today, the JDS has its detractors. Some behavioralists question the model's predictive ability (see Steers and Mowday, reference 3 in the bibliography). However, the behavioral scientists agree concerning the model's ability to measure motivation and job satisfaction.

Description of the Survey Instrument

The Job Diagnostic Survey contains 94 questions and produces information on 45 variables. The survey is administered individually, in approximately 30 minutes.

Most survey questions ask participants to record their response on a scale of one through seven. The following question is an example:-

6. To what extent does doing the job itself provide you with information about your work performance? That is, does the actual work itself provide clues about how well you are doing - aside from any "feedback" co-workers or supervisors may provide?

1-----2-----3-----4-----5-----6-----7

Very little;
the job itself
is set up so I
could work for-
ever without
finding out how
well I am doing.

Moderately;
sometimes
doing the job
provides "feed
back" to me;
sometimes it
does not.

Very much;
the job is set
up so that I
get almost
constant "feed
back" as I work
about how well
I am doing.

Core Job Dimensions

The H/O model identifies the three "critical psychological states" associated with high levels of internal motivation, satisfaction, and quality of performance. These psychological states are 1) experienced meaningfulness of the job; 2) experienced responsibility for outcomes; and 3) knowledge of actual results. The existence of these psychological states in a job leads to low absenteeism and turnover and high levels of internal motivation, satisfaction and quality of performance.

Five characteristics of the job (core job dimensions) are analyzed via the survey instrument. Skill variety (tasks that challenge the individual's skills and abilities), task identity (completing a "whole" and identifiable piece of work) and task significance lead to experienced meaningfulness of the job. Task autonomy leads to experienced responsibility for outcomes on the job. Finally, feedback concerning effectiveness of the employee's efforts provides a knowledge of results on the job.

Table 1 compares employee perceptions of the five core dimensions in jobs of data processing professionals versus other professions (from the Hackman/Oldham studies of over 6,000 persons).

Table 1.

Comparison of Core Job Dimension Between D.P. and Other Professionals.

Core Job Dimension Factors	Data Processing Professionals			Other Professionals
	Analysts	Prog/Analysts	Programmers	
Skill Variety	5.73	5.41	5.16	5.36
Task Identity	5.59	5.44	5.07	5.06
Task Significance	5.81	5.82	5.45	5.62
Autonomy	5.45	5.63	5.19	5.35
Feedback from Job	5.20	5.06	5.10	5.08

The Effect of High Core Dimensions

Hackman and Oldham use the three "psychological states" experienced by a golfer to illustrate the effect of core dimensions. "Consider, for example, a golfer at a driving range, practicing to get rid of a hook. His activity is meaningful to him; he has chosen to do it because he gets a 'kick' from testing his skills by playing the game. He knows that he alone is responsible for what happens when he hits the ball. And he has knowledge of the results within a few seconds."

1. Experienced meaningfulness. The individual must perceive his work as worthwhile or important by some system of values he accepts.
2. Experienced responsibility. He must believe that he personally is accountable for the outcomes of his efforts.
3. Knowledge of results. He must be able to determine, on some fairly regular basis, whether or not the outcomes of his work are satisfactory.

If these conditions exist, a person "tends to feel very good about himself when he performs well." Those good feelings motivate him to try to continue to do well. This is what the behavioral scientists mean by "internal motivation" - rather than being dependent on external factors (such as incentive pay or compliments from the boss). The J.D.S. model computes a single summary index which indicates the "Motivating potential" of a job. The index is called the motivating potential score (M.P.S.).

The relationships between the three psychological states and the on-the-job outcomes is illustrated in Figure 1. When all three are high, then internal work motivation, job satisfaction and work quality are high and absenteeism and turnover are low. A number of studies by Hackman and Oldham have substantiated the conceptual model in Figure 1. One was a study of more than 1,000 employees working on about 100 diverse jobs in more

than a dozen organizations over a two-year period. The effect on absenteeism and job performance was substantial, as shown in Figure 2.

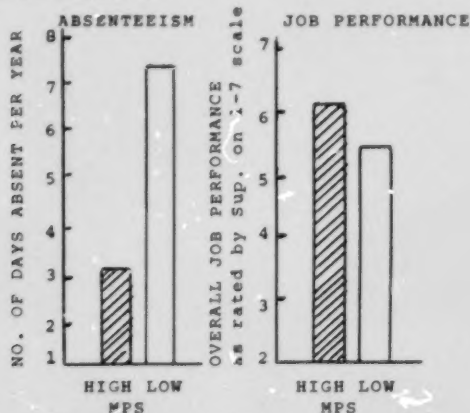


Figure 2. Absenteeism and job performance for employees with jobs high and low in motivating potential.

Table 2 contains the model output on psychological states, comparing data processing to other professionals. The results are comparable for "experienced meaningfulness." In the other two categories, d.p. professional scores are lower. The survey revealed the specific reason for the lower rating on "knowledge of results." It was all due to the category "Feedback from supervision." Feedback from the job itself was approximately the same as for other professionals.

Our supervisors are not doing anywhere as good a job (3.94) as their peers in other professions (4.21) in providing feedback to employees. This is an area where an immediate improvement is possible.

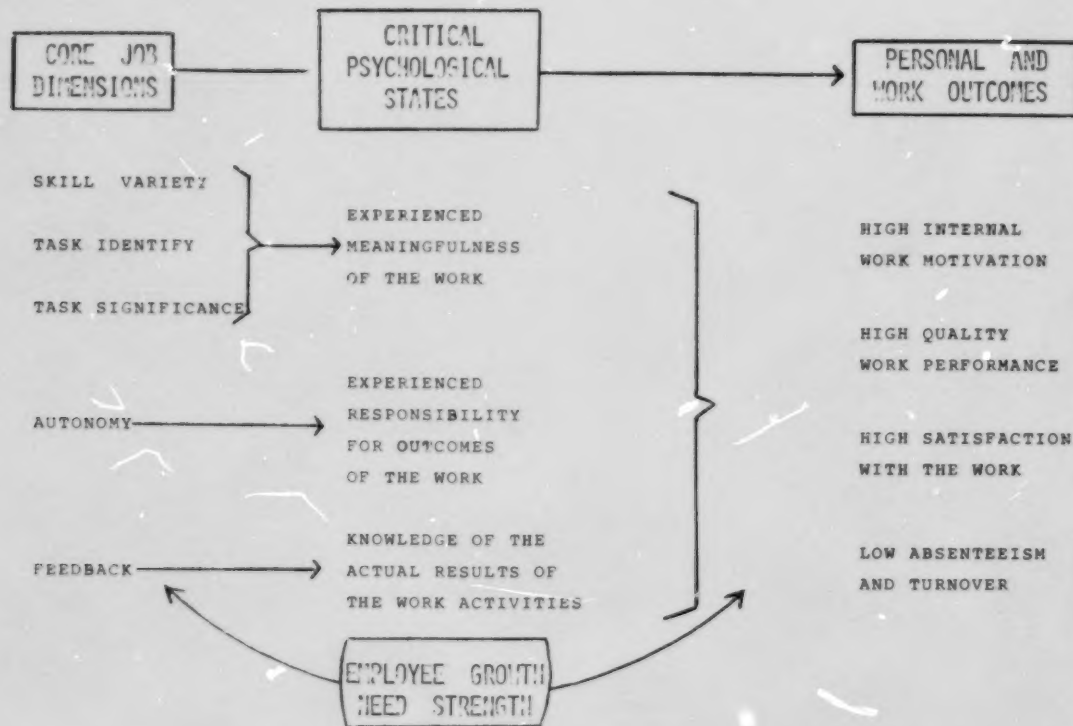


Figure 1.
The Job Characteristics Model of Work Motivation.

Table 2.

Comparison of Psychological States Resulting From Core Job Dimensions.

Psychological State	Data Processing Professionals			Other Professionals
	Analysts	Prog/Analysts	Programmers	
Experienced Meaningfulness	5.66	5.50	5.30	5.40
Experienced Responsibility	5.38	5.51	5.16	3.75
Knowledge of Results	4.60	4.43	4.40	5.00

Measures of Satisfaction

The right hand side of Figure 1 pictures the "personal outcomes" from an individual's job. The J.D.S. does not measure actual work outcomes: productivity, employee perceptions of their productivity, turnover, or absenteeism. Employees report directly how satisfied (or dissatisfied) they are with various aspects of their job. Table 3 provides those results. It indicates that d.p. professionals generally are satisfied (recall that the midpoint of the rating scale is 3.5). Their ratings are similar to those of other professionals.

However, in this category the J.D.S. proved to be an especially good discriminator. One organization surveyed was not "healthy", in behavioral terms. All others were. In fact we attempted to survey only organizations whose working environment was healthy, because we wanted to build "norms." In the past our profession has been without benchmarks in the behavioral areas.

The unhealthy organization was significantly below the norms on most of the core job dimensions. With the information from the J.D.S. we have a substantive basis for beginning the job redesign process in that organization.

Table 3.

Comparison of Satisfaction in Data Processing and Other Professionals.

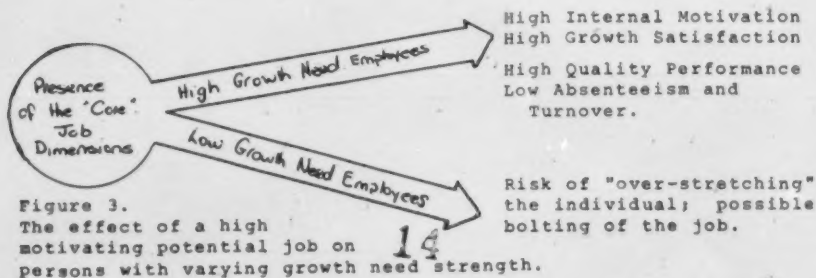
Measures of Satisfaction	Data Processing Professionals			Other Professionals
	Analysts	Progr/Analysts	Programmers	
General Satisfaction	5.19	5.34	5.17	4.88
Satisfaction with Co-workers	5.34	5.52	5.38	5.48
Satisfaction with Supervision	4.75	4.53	4.77	4.89

Growth Need Strength

The expectation is that people who have a high need for personal growth and development will respond more positively to a job high in motivating potential than people with low growth need strength.

Obviously, not everyone is able to become internally motivated - even when the motivating potential of the job is quite high.

Behavioral research has shown that the psychological needs of people determine who can (and who cannot) become internally motivated at work. Some people have strong need for personal accomplishment - for learning and developing beyond where they are now, for being stimulated and challenged, and so on. These people are high in "growth need strength" (G.N.S.) Figure 3, diagrammatically, shows how individual growth needs affect response to a job high in motivating potential.



Growth need strength is quite high for d.p. professionals, compared to other professionals and to other job categories. Table 4 shows this effect. This outcome has little surprise content for managers used to insistency by their staff that they be provided training, be allowed to attend conferences and seminars, etc. A frequent result of behavioral research is that intuitive beliefs are substantiated. Survey data on G.N.S. is a good example.

Table 4.

Comparison of G.N.S. and M.P.S. By Job Category

Job Category	Growth Need Strength	Motivating Potential Score
D.P. Professionals	6.14	162.1
Other Professionals	5.59	153.7
Sales	5.70	146.0
Service	5.38	151.7
Managerial	5.30	155.9
Clerical	4.95	105.0
Machine Trades	4.82	135.8
Bench Work	4.88	109.8
Processing	4.57	105.1
Structural Work	4.54	140.6

However, the key reason for computing growth need strength is to compare it with the job's potential to fulfill G.N.S. A job low in motivating potential will frustrate a person with high growth need strength. It is a perfect example of the old cliché of a round peg in a square hole.

The motivating potential score (M.P.S.) shown in the right hand column of Table 4 enables a comparison of the job potential to the employee growth need strength.

The M/O survey results include an example of the imbalance of G.N.S. and M.P.S. Notice in Table 4 that the lowest G.N.S.

is for structural work. On the other hand, the M.P.S. for that field is 140, which is near the midpoint for all jobs reported in the table. Jobs in that industry have a motivating potential above the growth needs of the workers. The lower prong on the two-arrow diagram in Figure 3 probably illustrates this situation.

On the other hand, the upper prong on that figure illustrates the situation for the d.p. professional. Both the individual's growth need strength and the job's motivating potential score are high - the highest in the table. As we will discuss in part two of this series of articles in Datamation, such a balance is not so prevalent among the other jobs in the data processing organization.

Likewise, although the average for all organizations in our study showed a high positive correlation between G.N.S. and M.P.S., this was not always the case for individual organizations. For example, the organization with the next to the highest G.N.S. (6.20) had the lowest M.P.S. (130.2). In contrast, another organization with the lowest G.N.S. (5.73) had the highest M.P.S. (209.8). Now that they are aware of this disparity, these organizations can analyse their jobs to improve the inequity.

Social Need Strength

The most surprising result of the survey was the measurement of the variable labeled "social need strength" (S.N.S.) Survey questions related to this variable determine an individual's need to interact with others. The average score on this variable for all other professionals was 5.65.

For all data processing professionals in our survey, the score is 3.70; for five organizations the average was only 2.23. While some professions attract people who have a high propensity for, and reinforcement from, interaction with others - our profession does not appear to exhibit this characteristic.

The implications are significant. Programmers and analysts have increasingly been grouped into teams in anticipation that productivity will be increased. The Chief Programmer Team (C-P-T) concept has been widely publicized and advocated.

Some have suggested that it is the social interaction of the team that has produced whatever productivity benefits resulted from the C-P-T approach. If our study is representative, it mandates caution in accepting such views.

What does this survey statistic mean to a manager of d.p. professionals? The lack of need for social interaction does not mean that teams should not be utilized. It does indicate that d.p. professionals are not actively seeking a team experience. Managers of other parts of the company will find their employees eager to interact - and a team approach is a natural organization to facilitate such interaction.

This conclusion is supported by data from Table 3. D.P. professionals in the survey organizations were quite satisfied with their co-workers. In other words, they were getting the interaction they needed - so long as it was not overdone.

Our industry, like others, has a number of experientially derived theorems - that have not previously been validated by statistical studies. For example, how often have you heard the comment that programmers are "loners?"

The G.N.S. section of our study lent authenticity to one of those theorems. Here is another case. Perhaps the theorem is best expressed by Gerald Weinberg in his widely quoted Psychology of Computer Programming: "If asked, most programmers probably say they preferred to work alone in a place where they wouldn't be disturbed by other people."

However, we interpret the survey results on social need strength as follows: management does not need to reduce emphasis on the project team, it just needs to control the frequency and duration of team meetings. Consider the oft cited guideline that structured walkthroughs should be limited to no more than two hours. You don't find similar guidelines in the literature on management meetings! (The social need strength for managers in the H/O surveys was 5.65) The structured walkthrough guideline was empirically derived - probably by judging the increasing level of impatience by analysts and programmers when a meeting dragged on.

Also, a low social need may indicate the need for additional training on supervisory techniques when d.p. professionals are promoted to management positions.

Summary of Findings

Although we have barely begun the in-depth analysis of our data base, some tentative propositions are:

1. D.P. Professionals Have Some Unique Differences From the General Population

- a. They have substantially higher growth need strength than any of the job categories surveyed by Hackman and Oldham. This is true of analysts, programmer/analysts, and programmers.
- b. They have the lowest social need strength among professionals - significantly lower.

2. Good Job Match is Possible.

For the survey firms - intentionally selected because they are healthy - the motivating potential of the jobs match the growth need strength of d.p. professionals.

3. Supervisory Feedback to Employees Can Be Improved

Employees in the survey firms are generally satisfied - with co-workers and with supervisors. Nevertheless, they believe that feedback from supervision should be improved. They rate this category lower than do their counterparts in the other professions.

4. An Appropriate Conceptual Model Exists.

The Hackman/Oldham model is applicable to the computer field. Our analysis substantiated that of the original researchers, with comparable statistical reliability and validity. The conceptual model (in Figure 1) is useful for management in our field to analyze individual motivation patterns and the Job Diagnostic Survey is useful to gather data to determine, satisfaction levels and work outcomes.

5. Job Redesign Has Potential

Studies of other industries by H/O and others (i.e. Rosenbach) show that job redesign can increase satisfaction and productivity. With the norms resulting from our study, managers in the computer field have the basis for determining which jobs have potential for improvement.

Conclusion

Productivity consists of two parts - improved techniques and increased motivation to utilize those techniques. An enormous amount of time and energy has gone into the first part - technique improvement. Why has the second part - motivation - been given so little attention?

Is it the qualitative nature of that part? Are we so oriented in the analysis of the quantitative part of the job that we willfully ignore the harder-to-measure aspects of employee motivation?

Or is it the fact that we are "systematic" people - who deal in practicalities. We want something we can touch or see. "Lines of code per hour" is a measurement that can be substantiated. Degree of motivation cannot be seen, felt, tasted or smelled - but it exists.

We are emulating the industrial engineers of the 1940's, who kept chipping away at each job on the auto assembly line until it was splintered into the ultimate level of specialization. We seem to be concentrating just as fervently on fragmenting the jobs of analysis and programming. The Chief-Programmer-Team concept is the latest in the long list of activities designed for enhancing specialization. The analysis/programming task now is fragmented into the elements performed by the chief programmer, the moderator, the librarian, the recorder, etc. Perhaps jobs should be enlarged in scope, rather than reduced.

If we devote "equal time" to the analysis of motivation and the ingredients of job satisfaction, this alone may increase productivity. However, productivity increase is the wrong reason for initiating such a study. Shouldn't a supervisor gain as much satisfaction from helping employees achieve fulfillment as from meeting cost/schedule objectives?

Fortunately, researchers like Hackman and Oldham have proved in other industries, that productivity can be increased by a better matching of jobs with individuals.

We already have a data base rich with information for further analysis. For example, our data on problem areas and on compensation preferences is being analysed for our next Datamation article.

During the next year we will be increasing our data base to include more than 2,000 professionals, to insure a representative set of "norms." At the same time, we will be conducting job redesign studies, to better understand the factors which produce job satisfaction.

Our profession has been very progressive in developing new "product-related" techniques. Let's hope that progressive attitude will now extend to better utilization of the behavioral sciences.

KEYNOTE ADDRESS

"ERUA NGA WAAHI O TE PA"

by

DR BRIAN COX

Dr Cox, Dunedin born and bred, has been very actively involved in the Data Processing field both in New Zealand and overseas.

After graduating in mathematics from Otago in 1959, Dr Cox won a commonwealth Scholarship and studied at Cambridge for three years, and on his return to Otago was appointed Lecturer in Mathematics. During this period his research work covered Fluid Mechanics and Limnology.

In 1966, Dr Cox was appointed Director of the University of Otago Computing Centre, and since then his activities in the computing world have been many and varied. He has held several visiting lectureships and professorships at notable universities overseas, and was a Nuffield Travelling Scholar in 1971.

His activities in the New Zealand Computer Society began in 1964 when the Otago Branch was inaugurated, and since then he has been active at Branch level, played a major role in the 1970 National Conference, was a National Councillor from 1967-1975, and was National President in 1974.

Dr Cox was an IFIP Delegate in 1974-75, and as a fitting reward to his efforts was created a Fellow of the New Zealand Computer Society in 1976.

Mr Chairman, Ladies and Gentlemen,

Some four years ago, I attended a large computer conference, IFIP74. After four days of proceedings it was not easy to recollect much more than overall impressions. But one talk, by Kleinrock on Queuing Theory, does stand out because it had a simple theme of "bigger is better".

Hence when preparing this talk I decided to choose a specific theme in the hope that in due course you would at least remember the theme if not the finer detail. To make a particular point the theme is expressed as the title in Maori "Erua Nga Waahi O Te Pa" which is a direct translation of "Communication is a two-way process". (Though it is also capable of much aliterative extension as happens frequently and often beautifully with Maori.)

When I started preparing this talk I had hoped to find a Maori proverb for a title, and spent a fascinating day in the Hocken Library looking for one. I found several that could apply to our industry, e.g.

Kore te hoe, kore te taataa
- alas without paddles or baler

or the cry of the users associations

Nga korero o era rangi, mahue noa ake
- promises of other days wholly left behind (or never trust to fine promises)

or the delightful

He hiore tahutahu
- an often signed tail (a description of a lazy dog staying by the fire despite the height of the flames)

but none that could be bent to describe the theme of communication.

For the moment let us ignore the technological aspects of communication and consider it in its meaning of a social process. In its simplest form, communication consists of

a sender, a message, and a receiver. This applies equally to mass communication like radio, TV and newspapers, as it does to interpersonal communication by speech, signals, gestures, pictures, graphic arts, literature and music. The interesting thing is that the message only has such meaning as the sender and receiver's experience permits him or her to recognise, or if you like, we can say that a person can communicate only in terms of his own frame of reference.

For example if a primitive man had never seen or heard of an aeroplane, and one were suddenly to appear, he would probably think of it as a great bird, and the pilot as a supernatural figure able to tame gigantic birds. The same person, confronted with a computer is unlikely to have a frame of reference in which to place it and might therefore dismiss it as unimportant. As a different example, a New Zealander teaching in Africa noted that the first few times she called the roll of names in her class, the students laughed, and she worked desperately hard to learn to pronounce the names correctly, because she thought they were laughing at her unskillful way of saying their names. But when she became skillful, they laughed all the harder. Finally she learned that they were laughing out of friendliness and pleasure at how hard she was trying and how well she was doing. In her frame of reference, a laugh at that point meant derision; in their frame of reference it meant applause.

The message passing between the sender and receiver has other interesting characteristics, for example surface and latent meaning. Think of "Good morning" next time you say it on a cold wet morning to a close friend. Messages also often consist of a number of parallel messages. This happens particularly with verbal messages, but applies also to written messages in which type face, size of print, quality of paper, white space and other factors influence your acceptance of the message.

The sender and receiver also have great influence on the message. How often have you felt that a message, even a simple one, has been distorted, misinterpreted or rejected. One piece of research has shown that people who are strongly prejudiced on a certain subject will often interpret an attack on their prejudice as praise and support for their viewpoint. Other studies have shown that feedback from the receiver is also important, but also subject to misinterpretation. Remember that communication is a two-way process. If we look we find the concept of sender-receiver also recorded by the Maoris in that the two great pillars of a meeting house are called te arero (the tongue) and te taringa (the ear).

So far I have been talking about the simplest unit of communication - the two-person group of sender, message and receiver. Communication in society is made up of complicated networks and long chains of senders and receivers. An important characteristic of such chains is that every person in the chain except the first and last is a gatekeeper. He can pass on the message, alter it, add to it, remove part of it or even suppress it. Each gatekeeper has great power over the message and over those further down the chain.

Communication is essential to society because of its important functions, which are:

- our need for information
 - our need to make decisions
 - our need to store accumulated knowledge
 - our need to socialise (or educate) new members of our society (to give them the same frame of reference)
 - our need to communicate entertainment
- and perhaps the most important of all
- our need to maintain everyday relationships of human beings who must live and work in close proximity and adjust to each others needs and quirks while maintaining an efficient and rewarding existence.

Because these are important let me take them in turn.

Let us look at aspects of the role of computers in New Zealand in each of these.

Our Need for Information

The use of computers in analysing our environment is obvious, particularly in government departments like Electricity, Reserve Bank and Railways. There are also many research projects on subjects like solar energy, wind energy, rainfall and irrigation as well as on socio-economic studies of hydro development. No doubt we shall see more and more use of computers for such studies. However as we come to depend more and more on computer models, we depend more and more on "expert" opinion - the opportunity of concerned groups to interact with the model and to interpret the results themselves. They do not have access to or finance for computer models.

There is much that we could do to develop files of public information, and to make them available to everyone, and not merely to experts. In doing so we would need to be conscious of the privacy of some data, and would need safeguards to protect it. To assist the public have access to public information is a direct application of the code of ethics of our Computer Society. Do we want to do it? How do we do it?

We need to remember that information is valuable, that timely up-to-date information is even more so, that the ordering of information may further enhance its value, and that information collected for one purpose should not be used for another without permission of all involved. I was disappointed to find during a controversy in Dunedin earlier this year that these aspects were not properly understood by most people.

Our Need to Make Decisions.

Professor Gottlieb said to an annual general meeting of our

society a few years ago - "information is power". All firms, organisations, societies, governments need to make decisions. They must or they will not survive.

Computers can assist the process of decision making, but we need to be very aware of the risks inherent in depending entirely on computers and on those who have access to them. If this is a direction in which society as a whole wishes to move then perhaps we should accept it. But we should be on guard against the development of exclusive ways of making decisions and to have a system of checks and balances to ensure that such power is not misused.

Our Need to Store Accumulated Knowledge

A byproduct of the routine use of computers is the ability to accumulate information - often at embarrassing speed as anyone involved in a tape or disk library would confirm.

However we have been slow to combine the two major repositories of knowledge - computers and libraries, and developments in other countries are now well ahead of us. With some few exceptions, we are not making wide spread use of automated bibliographies, of selective dissemination of information, of information retrieval services. Now many of these need to be international to be effective, and it may be that it will be more economic to pay for access to such systems rather than to develop our own. Most libraries in New Zealand will take their lead from the National Library which would need additional resources to develop such facilities.

I believe we shall see major developments in this area within five years.

There are other interesting examples of accumulation of knowledge in New Zealand. For example one of the first applications of computers (in 1964) in New Zealand was an analysis of all recorded cases of leukemia by location and time. This was because New Zealand is small enough and

homogeneous enough to have very complete records in many areas. A further and on going example is the data on the adverse reactions to drugs which is part of a wider collection by the World Health Organisation.

It is interesting to note that a further file of information, the National Poisons Register, is not held on computers because it needs to be available from anywhere in New Zealand twenty-four hours a day seven days a week. At present the only way of providing such a service with complete reliability is by telephone and card index.

Our Need to Educate

Computers have become an integral part of education, but not to the extent that they should have done.

Taking each of the main areas of education in turn.

At the primary level, we see little application of computers, and should not expect to, except as a minor adjunct to existing work. Without getting involved in the "Back to Basics" discussion, there are some fundamental things that children should learn at that level, which could include an appreciation of what computers do, but only to reinforce other concepts. It is more important that time is spent on reading and comprehension, writing, and arithmetic (with and without calculators!).

At intermediate level, there is more scope. I have been involved with some of our local intermediate schools and have been interested to see how quickly children of that age can grasp certain concepts. In fact a weekly class "in club" time for a term can cover a lot of useful material and even some programming - of simple calculations to reinforce their mathematics work, and of alphanumeric data to demonstrate wider concepts. However such activity must remain a modest part of the term's work.

At secondary school we can do much more. Some schools in New Zealand have their own mini computers, and with steadily

steadily decreasing prices many could now afford a micro computer. In addition many schools have links with friendly computer organisations, or make use of national systems like that described in this morning's paper by Collins and Hunt.

Compared with five years ago, there are now many teachers with some knowledge of computers and computing, but it is by no means universal. This is partly because our secondary system is largely driven by the syllabi of national examinations in the Fifth, Sixth and Seventh Forms. At present only one subject, Additional Mathematics, has a computing content. This subject is at the Seventh Form level, and only attracts a minority of secondary school pupils.

To make computing more widely known and understood we probably need a Fifth Form level subject, preferably compulsory, possibly called Liberal Studies, containing a section on computers, their applications and implications. What material are we prepared to do without to make room in the school year for it?

At the tertiary level we have three types of institution - Teachers Colleges, Technical Institutes, and Universities. From the point of view of computing resources, the Teachers Colleges have virtually none (but usually have access to neighbouring universities). The Technical Institutes have very little, and of the three the Universities are the best, but still inadequately equipped. In fact taken as a whole, the computing resources provided for all educational institutions (schools, colleges, institutes, and universities) over the last five years would be less than the cost of single installations in industry or government.

Despite this, we have seen developed the TCA Certificate in Data Processing, courses in Computer Science at all universities, course in Information Systems (i.e. the commercial uses of computers) at several universities, and considerable interest by students in these courses - to the extent that it has become difficult to provide the resources to support such

courses. In fact, I gather that in 1979 some universities will have to restrict entry to some Computer Science courses for lack of sufficient resources. That this should happen at a time when the industry is desperately short of people is nothing short of a tragedy.

There is one aspect of education that to me is very important - continuity, i.e. the ability to select a course and see it through to the end. In my experience, the universities, when changing regulations, go to considerable lengths to protect the interest of students. Unfortunately this does not apply to the Technical Institutes, as the following example will show.

When the Society supported the introduction of the TCA Certificate in Data Processing some years ago many of us were pleased to note that if sufficient students presented themselves to any Institute, a course had to be provided. Perhaps in the larger centres it has worked this way, but in the smaller ones it has not. It appears that the regulations governing Technical Institutes require that courses have an average attendance of 11.7 students. If, during the year, the attendance drops, the average of the classes may fall below the mystical number. The Institute may then have to cancel one or more courses to restore the balance. This it can do during the year. Hence your ability to complete such a course is dependent not only on your ability in the subject, but also on the stamina of your class mates. I find this incredible. Notice also that has a ripple effect in preventing subsequent courses from ever getting off the ground.

Now defenders of the system point out that there are Technical Correspondence courses and block courses at the Central Institute of Technology to cater for those from areas where numbers are small or uneconomic. While this may be true, the reality is that not all courses are available by correspondence and not all students (or employers) can afford to go to block courses in Wellington.

I believe we can improve this system. Courses, once started, must complete the year, even if reduced to one student. While I would prefer to see no minima for such courses, I would accept a modest number, like five, provided it applied only once, at the beginning of the year so that there is time for alternative approaches for any students affected.

Our Need to Communicate Entertainment

The role of computers in entertainment is less well established, but in recent years there have been interest in the printing industry, and use of control circuitry in such diverse locations as lighting systems for stage plays and satellite tracking systems.

Turning this topic around, there is the possibility of computers being used for entertainment, and I am sure we are all familiar with games like Startrek, even if we do get "wiped out".

What I fear also happens is that computers are used by entertainment. I have noticed what I fear is a growing trend for the media to sensationalise and dramatise computers. For example last year when we were holding the data communication seminar, one of our TV channels tried hard to get a sensational or newsy item out of it. They were not interested in practical or useful applications. Nor I fear was this an isolated example.

As another example, a current affairs radio programme often involves opinion from more than one person. I wonder whether you realise that these "discussions" are prerecorded with questions being asked by the producer independently to each person, and the resulting tapes being edited. The people assembling such programmes have an awful responsibility - they really are "gatekeepers", with profound ability to influence the receivers of the information.

As a slight aside, studies in communication have identified

a particular aspect and called in "cognitive dissonance". Some examples to describe this are

A student who has studied very hard for a relatively unimportant examination, may tell others when an important examination it really is.

A parent who sends a child to a private school may overwhelm others with his description of the advantages of that particular school.

How many of you, when buying a new car, suddenly see many other cars of that type on the road, and advertisements for them wherever you look?

One view of this is that a decision on alternatives often causes dissonance because of attractive features of the rejected alternative. We then tend to reduce this dissonance by conscious and unconscious means to reinforce our decision.

When viewed in this light it is not at all surprising that a failure in a computer system is seen as power supply problems by an engineer, as hardware by a programmer, as software by a control clerk, as lack of resources by a manager, and human stupidity by an outsider.

It is my contention that the sophistication of systems installed in recent years has outstripped our means of communication about them (not with them, but about them). Some examples may make this point of view clearer.

Our early systems were predominantly mono programmed batch systems. Users came to a centre with their work, implying frequent contact with the providers of the service. If problems arose they were usually the only ones affected. With relatively few users, communication with all was easy. Most could be given a message by telephone in less than an hour.

With the use of multiprogramming we had to be more careful about the influence of one user on others. (I remember well the day when one of my research students demonstrated how easy it was to write binary zeros or any other pattern all

over the operating system. The same person later showed me how two paper clips could open any door in the building!! Multiprogramming brought more complex operating systems, and the need for users and operators to learn how to communicate with them. For many users, interested primarily in getting their task done, this need to learn other tricks created considerable dissonance. Faults became always those of the operating system, never in the users program.

With the development of networks and remote access, communication within the system improved, but with the system did not. It became all too easy for remote users to become isolated, to be less well informed, to be without independent means of communication. (I know of one system being installed a few years ago in which the frustration was substantial because when testing at night for problems with the telecommunications link, the nearest independent telephone was a public call box over a mile away - hardly conducive to good communications. I can also recall one university concerned about communication who installed an independent set of lights beside each cluster of terminals to display green (all systems going), amber (temporary failure - back soon) and red (serious failure - go away and try again later), and another university, less concerned, who had a machine room switch to put a busy signal down all lines to prevent users enquiring on status after a breakdown.

New Zealand has particular problems in this regard because our main centres are stretched out in a long thin chain, rather than clustered together for collaborative warmth. To make it more difficult there is a stretch of water across the cañon making personal communication much more difficult and more expensive.

As a result we have to rely more than some countries on written communications. Now written communications have their own problems. Consider for a moment the inscription "Little Wonder Fuse Blower : Insert in mains socket and PRESS! "

or the phrase "TIME FLIES LIKE AN ARROW" for which there are several distinct meanings. The nuances of the English language add richness but also ambiguity. How often have you misinterpreted a phrase in a letter to the extent of subsequent embarrassment.

If we add the fact that much of our technical literature is imported, often with sufficient delay to make it less than relevant, we also risk introducing a cultural difference. To avoid this, the majority of technical literature discusses syntax and semantics only in positive terms, i.e. the meaning of valid constructs, and does not even attempt to describe what will happen if you do not follow the instructions. To do so in detail would result in large unwieldy descriptions.

This is one area where terminal systems could have a tremendous advantage. Any self respecting user orientated terminal system has a prompting or "help" facility to assist new and experienced users. This facility could be invoked by the system if certain types of error are detected. It does depend of course on high reliability and integrity of the system providing it.

Another aspect of communication is the effect of time. One of Parkinson's Laws describes the art of decision making as the postponement of decision until it is no longer relevant. (e.g. delay buying a warm jersey until late spring when you no longer need it) - The art and science of procrastination, and this is relevant to several aspects of computing.

For example, the standard software supplied with most systems is imported. If errors occur (they never should, but they do) the originators require sufficient documentation to make the problem understandable in their frame of reference, and with delays even to airmail, response from any other country in much under a month is optimistic. A thorough original investigation usually supplies a detour to the problem and may provide a solution. Hence a degree of self-sufficiency

is essential if you need to avoid the delay of communication in a long chain back to the originators. This lack of prompt communication contributes to reinvention of the wheel, but a better view is that self sufficiency could be the spare wheel which keeps you moving and operational.

A further effect of time arises in the daily organisation of co-operative activity. Anyone responsible for the work of others must provide them with clear instructions. It is however as important to the manager to know when a task is complete and whether problems have arisen in the process so that he or she can place the activity in the overall task. It may be a sweeping generalisation, but many programmers are not good at reporting back, and projects suffer as a result. Remember - Communication is a Two Way Process.

In my introductory remarks I suggested that communication was important from our need for information and from our need to make decisions. Government in New Zealand, both local and central, have been quick to realise this, and have installed many computers to assist the process of government, to the extent that our society has become dependent on computers and on communications. In 1971 I was in the United Kingdom at the time of a major postal strike. It was interesting to note that because the automatic telephone system continued to function, communication remained possible, and sever dislocation was avoided. Some interesting alternative techniques for transmission of essential written material developed from the dislocation to mail services. It is salutary to reflect on the effect of a complete breakdown of power supplies for a week or more would have on this country. It would mean no telephones, no mail, few computers, and few decisions. The price would be enormous, but the pleasure fantastic! What would be the effect of partial collapse of the system?

Are we using computers enough to help with making decisions?

Do we have information retrieval services that assist with the development of appropriate legislation? Do we use existing technology to ascertain public opinion on major issues? Do we use technology to produce accurate and unbiased information for the public via the media? I fear that the answer to all of these questions must be "no" in general - though there are some isolated exceptions.

It is for these reasons that I suggest that our sophisticated systems have outstripped our means of communication about them. What should we do about it? Some of the more obvious approaches are:

1. Improve interpersonal communication

This requires a greater awareness of the needs of communicating individuals, improved working conditions (tired people do not communicate well), improved understanding (would that humans would give an ACK or NAK like a data communications transmission).

I have recently enjoyed immensely a book entitled "Most Secret War" by R.V. Jones which describes the development and application of scientific intelligence in World War 2, particularly navigation, radar, and effective counters to new weapons. The book was interesting for two particular reasons. Firstly, it describes a process of reasoning which I find extremely similar to the process of locating a fault in a computer system.

(As a lengthy aside, I found the reference to Occam's Razor "Essentia non sunt multiplicanda praeter necessitatem" which when translated from the Italian version of Algol68 is "hypotheses are not to be multiplied without necessity", directly applicable to some recent problems on our system. For, if you start allowing more complicated hypotheses than are essential to fit the facts, you can launch yourself into a realm of fantasy where your consequent actions will become misdirected. As a light hearted example of this I could explain that the reason I never see a Kiwi in my machine

room is that although there really is a Kiwi there, it is very shy, particularly of me, and when it hears me coming will hide under the false floor! Next time you are looking for a bug in a program, try using Occam's Razor - multiply the facts not the hypotheses.)

The second reason for my interest in the book was because of its description of communication in various ways, and the importance of human communication. For example, I quote from p. 505.

"The premium thus placed in a modern scientific career upon undue concentration at any early stage encourages the production of men (and let me interpolate "women") with a less than average appreciation (for their standard of intelligence) of the world at large. This ignorance may even become a habit in after life, when the energy of the hothouse forced scientist declines, or when, finding himself outmaneuvered in his first few clashes with administrators, he retires embittered into his laboratory and decides that his ignorance is really a virtue since the worldly-wise are such rouses"

You can obtain an equally accurate statement by replacing scientist by programmer and laboratory by terminal room. We are running the risk of reaching short term goals, but paying a shocking long term price.

In my frame of reference the term "workaholic" is not a congratulation, but an insult. This is not to say that I encourage laziness or that some single minded people do not achieve great results. I don't - and some do. But individuals, organisations and society can pay a high price.

The time has now come when we should make sure that computer professionals are educated for the problems we have yet to solve, rather than trained to solve the problems we do understand. As part of this education we should include a substantial understanding of communication skills and so lessen the introspection, the withdrawal, the egocentricity, and the lack of humanity into which we can so easily fall.

Time and attention spent on personal communication can be warmly repaid, not merely in the better understanding of the subject under discussion but also in the human warmth of the discussion.

2. Improve methods of communication

A universal STD telephone system would be an advantage and will reach much of New Zealand presently. Improved mail services, facsimile transmission, electronic mail, would improve our ability to transmit written documents. From time to time I watch the way users of our university terminal network send messages to each other. There are times when personalities appear on the screen as well as the bare message!

Consider for a moment the potential use of a message switching network attached to the telephone system. This would enable you to transmit a document directly to the office of the receiver and, if the receiver wished, to obtain a response. You could also "converse". Admitted this is not very different from TELEX, just much faster and universal. Some sort of mail box facility would be needed to avoid interruption and to give one privacy.

I sometimes wonder whether we are rather slow to use a new technology effectively. For example, are tape recorders used as effectively as they might be? This thought arose from the need to interview an applicant in England for a position at Otago. A colleague in England recorded the interview on tape, and added subsequent comments rather than writing a report. As a result we got a much more detailed report and a clear impression of the candidate. The only disadvantage was the two hours it took to listen to the tape.

How far can we combine the concepts of word processing with message switching? For me, letter and report writing has become a tedious chore. It is high time we had tools which made the process of getting a written message to one or more others much much easier.

ARPANET, to which we could also link - if we could afford the toll line costs.

Reliability is also a function of simplicity. There are times when it is better than a complex unreliable larger one. With cheaper hardware, this may well come about, and there will be less tendency to operate simplex systems around the clock. The latter is undesirable because if successful, reliance will be placed on such an operation despite its inherent fragility. If unsuccessful the operations of the firm are jeopardised. I suspect that many computer systems in New Zealand are fragile in the sense of being over-used or under-staffed, or both. It would be an interesting project to find out whether this was true.

4. Develop Better Tools

In 1970 Richard Hamming urged our conference to develop more tools. In the eight years since, how many good tools have we developed? How many are communication related? There are some, but mainly I suspect in the hardware area rather than in software. Yet with the implicit cost of software rising, and that of hardware falling which is the more important?

I hope that we, as professionals, are not the last to use word processing capabilities for our documentation, or to use information retrieval systems to be aware of the work of others in our field.

The development of effective tools is particularly important in New Zealand. While we have some substantial computers, the vast majority of our systems are small. We have also developed the practice (partly as a result of import controls some years ago) of driving our systems very hard.

With small systems supported by relatively few staff it is particularly important to save time, energy and money by developing (or buying) and using good tools. What tools you say? Tools like librarians, editors, packages that do your

job, structured languages that allow development of programs within ones intellectual control. They don't exist you say. Then why aren't we developing and exporting them?

It is possible that our national tradition of "do it your self" which applies to so much of our activity is contributing to the "lets do it our way" approach, and unnecessary duplication. It would be more to the point to develop simple robust tools and to communicate them to others. This will encourage them to reciprocate.

5. Be More Professional

Over recent years there has been a clear development of a professional approach to computing. If this were not the case we would not have successful systems, both large and small, in daily use.

But there is more we can and should do. For example we should be much better at estimating the completion date of projects. I know I have been involved in projects which have taken much longer than expected. I have heard most reasons for late completion and invented a few of my own. Many can be attributed to poor communication at one level or another.

We should also be prepared to make professional judgements and to stand by them firmly. If we were better at estimating we would know the resources we needed to complete a project. It would be unwise to start a project if in one's professional judgement the resources were not available to complete it in the specified timescale. There are times when a carefully assessed "NO" achieves more than a thoughtless "YES", and subsequent opinions will be more respected.

In conclusion let me remind you of the importance of communication to society because of:

- our need for information
- our need to make decisions
- our need to store accumulated knowledge
- our need to socialise (educate) new members of our society.
- our need to communicate entertainment
- our need to maintain everyday relationships.

Computers can assist, but only if we want them to do so, and make them do so.

You are the group that can make it happen, and when doing so remember

Erua nga waahi o te pa
 - communication is a two way process.

KEYNOTE ADDRESS"COMPUTERS AND COMMUNICATIONS"

by

PROFESSOR A.S. (SANDY) DOUGLAS

A Ph.D graduate from Trinity College, Cambridge, in Mathematics, Professor Douglas has held the Chair of Computational Methods at London School of Economics and Political Science since 1969.

His career has been extremely colourful and varied, and he has to his credit a number of important consultancies to world-wide organisations, including United Nations, Anglo-American Corporation, Shell, Phillips, I.C.I.

Since being a Founder Member of the British Computer Society in 1957, Professor Douglas has had a very active role on executive and sub-committees. He was President of B.C.S. in 1971, and has represented the United Kingdom on various International Committees, the most prominent of which were the I.F.I.P. General Assembly and his Chairmanship of the I.F.I.P. Committee for International Liaison.

Mr President, Mr Chairman, Ladies and Gentlemen I ought to begin this by thanking you for being here. When I was told I was going to address the last meeting on the Saturday I suggested that Graham and I should take to his boat rather early. However, I am very pleased to see how many are here and I must say thank you to all those people I've met in my travels around before the Conference and say how much both I and my wife and son have enjoyed visiting you all because it really has been a most marvellous experience.

I was a little afraid from what Peter Collins had said that the Society might suffer from Brucillosus. There are a number of Bruces around and as you know Brucillosus involves miscarriage of some kind and I suppose the buses must have gone wrong. I understand that's not actually the meaning of the word.

I want to start today by trying to get some idea of what I see the future of our business being. I suppose that I would like to see ultimately a sort of ideal terminal. It should accept dicatation in any language. It should accept written instructions, in any language. It should accept body language, including key strokes and allow access to any machine world wide to carry out work on the program. It should prompt users when they need it. It should display graphically in three dimensions. Put hard copy on demand in any typeface required, with multiple copies if needed, graphical in the form of drawings, figures etc, with page makeup. Now, that's very like a secretary with a word processor actually. Perhaps that's what I really need, not a terminal I could operate myself. Particularly as regards the body language you understand.

Well, how are we to achieve this ideal? In point of fact, technologically almost all the things I've suggested are either possible, or will probably be possible within a short time. Some things will be more difficult than others.

Possibly reading my hand writing has always puzzled my secretaries and I daresay you have the same problem. I don't suppose that the machine is likely to be anymore intelligent. On the other hand, it is possible and I have watched a demonstration of it in Sydney on my way through, to recognise signatures. Insofar as this represents a fairly significant advance in the direction of taking on handwriting we're getting very close to it. But the key of any system like I have suggested in which we essentially isolate the user from the supplier by a mechanism if you like, you start at the terminal and you can now connect yourself up to all sorts of computers anywhere.

The heart of this of course is telecommunications. I feel that perhaps in the conference in focusing in the man to man problems we haven't quite focused enough on the very real telecommunications problems that exist. The Post Offices are not exactly free agents. They have a great deal of your money already invested in the ground. They can't necessarily move away from this easily. They are expected by the tax payer, taking one year with another to make at least no serious loss. Despite the fact that imposed upon them very often are requirements for service in remote areas, which economically they probably couldn't justify.

They have a system which is essentially voice oriented to a very strong extent. It is also oriented essentially to connect Mr A with Mr B. It is not oriented towards connecting with Mr A with Messrs B, C, D and E. You have such a system, it's usually called television or something like that, connects Mr B, C, D, with anything up to X, Z through a broadcast system. But we don't think that way in the telecommunications industry. Perhaps we should. I think far more thought must be given to whether the actual needs of a community are the A to B situation we have now or whether we're more interested in various kinds of broadcast facility. It is implied in our technological development that the word processor which is, if you like, a built up

typewriter in fact, if this is connected to other word processors we can pass information from one to the other over the telecommunications networks and in this way we can avoid ever having to put out hard copy and put it through the mail. In fact, as you well know, in any office operation you very often wish to send several copies to different people on a distribution list and the present arrangements would mean that we have to send duplicate copies down to various other people; If we put in for instance, packet switching equipment, then it would be that much easier to do this. But we need equipment which will give this kind of thing.

At the instance of Dr Robinson I am very carefully avoiding telling the Post Office how to do their job and saying essentially that I think we should get a clearer view of the specifications of what we would like done. I think what we would like done in many cases is not necessarily what would be simple with the current equipment. I would like people to start thinking what they would really like and not what they would currently be able to get and to put these requirements on the appropriate authorities. The major difficulty we have found with our Post Office people is to convince them that demand exists for anything at all, other than what they are currently serving and that this demand could be met at an economic cost. In the first place, I remember many years ago now, when I was in C.E.I.R. they sent me a questionnaire around saying, Would we, if we had, could we use in effect, a system which they were thinking of putting out. At that time they were so vague about what they were going to tell me that it was impossible to discover whether I wanted it or not, but knowing the Post Office I immediately said what I wanted was in fact enormous numbers of fast lines all over the country at no cost. Since at that time we were owned by B.P. there was some slight credibility in the fact that we might be able to pay for it. However whilst I don't advocate that kind of

slightly wild approach to doing it, I do think that the Post Office needs honest figures from somewhere. As far as its able to. It's an extremely difficult problem. If you are given things at different tariffs when you will obviously use them more or less. Certain things are probably unusable. I gather an attempt to introduce DATEL 200 was not altogether successful because nobody actually wanted it. On the other hand, there are other things that are wanted because you currently have networks over the country and clearly there needs to be a long term plan related to this. One has to remember that the Post Office has to work in very long term.

One of the most important things that you have to deal with in the telecommunications field, having established that there is a need and having established that it can be met reasonably is to insist on some degree of standards. Now at the moment, standards take a long time to develop and its not particularly easy to do. However, we have got as far as one standard called X25 which is beginning to look like something useful, and in fact some PTT's, the German Post Office for example, now insists that everybody tendering within Germany uses this protocol if they are going to go across their systems. However, it's rather like language protocols with which you are familiar. The protocol is fine but the implementation is different in each case. We're still in the situation where it isn't entirely clear that the implementations will be in common between countries even and its not altogether clear what X25 means. Nevertheless, it is possibly the only hope for getting a reasonable standard at that level at this time and I would urge you to take very careful note of that fact and not go too far away so that you can't return to some standard based on it. There are of course, many other things than just the part that X25 covers. There are end to end protocols and a whole lot of other things which probably ought to be standardised for the good of all. I will read you a passage

from our own national interim report of the National Committee on Computer Networks. It says - "in the end standardisation always pays off for the user by giving him freedom and for the manufacturer by increasing his market".

I believe that to be true. I believe it also something which although possibly you may feel that you are a small enough community that you will not dictate the terms of the major manufacturers, nevertheless this is the situation where the dripping of water on the stone is a necessary process and everyone that does just their little bit towards putting standards on manufacturers, and indeed trying to adopt them themselves, does something that is useful to the community as a whole and I feel sure that we should remember that.

Well moving in standards towards languages we're moving away from the machine-machine problem to the man-machine problem. There is a worry among many people that if we standardise on language we, in fact, increase its rigidity. Of course, this is in part true. However, there is no particular reason why a language once standardised need to be totally rigid provided that developments are permitted within a fairly constrained set of rules and this has been shown by the Algol developments over the years, except I suppose, that one could argue that Algol 68 was an unfortunate sport. However, the fact of the matter is that the basic languages that are used in industry, therefore in 80-90% of all programming effort are Fortran and Cobol. This is a fact of nature and it is not in my philosophy to buck nature. Essentially if we want to move away from Fortran and Cobol it is no good teaching everybody how great your latest snowball or tiddlebol is. You've got to explain how somebody with a whole lot of programs in Cobol or Fortran is going to move away onto this new system. Of course you can do this well or badly. If you look at what has happened

in the data base management field we tend to have gone along some very complicated paths as outlined by the codicil proposals, which are an attempt, of course, to retain Cobol while at the same time doing things that Cobol never thought of. This has to be given that kind of consideration, but one does wonder if it has to be as complicated as all the documents seem to make out. Clearly more work needs to be done in the direction of getting us off the hook.

But let us direct the work in the right direction. The direction is to say we have these things as a fact of nature. What are we now going to do in order to avoid the problem. Well, to some extent of course, the advent of on line computing is changing this anyway at the Fortran end, particularly by moving towards Basic. Whether this is an advance or retard depends on your point of view. However, it is clear that the development of languages in that area will be constrained by certain other aspects of the thing. However, one wonders whether in those cases quite such formal languages are required if the terminal is itself capable of prompting you over many of the things that need to be done. Whether the rules cannot be made a good deal more flexible and, I think what one ought to be trying to do with that is to make use of the full power of terminals to make things easier for the user. I think this is a thing that we must remember to try and do.

In certain areas, progress has been made, of course, with this. The Department of Defence in America has produced a series of specifications starting with Strawman and going through, to what is now, Iron man, which is presumably in a slightly harder state than the initial propositions, which is intended to be a language for minis and micros which is intended to handle online response systems, particularly those related to radars and so forth, which the military are interested in. There has been a need for a language in this area for some time. The original language was called Jovial, which was used on the Sage system, an enormous defence system in the States. At a later stage

that developed, in the U.K., to a smaller language called Coral which has been standardised and is quite widely used through the U.K. defence system. The Iron man movement appears to be an attempt to bring this forward one further. Coral was modified into Moral, in attempt to achieve the interest of the Defence Department. But among the languages, interestingly enough, which were put forward eventually for implementation of Iron man, Pascal has found the greatest favour. Now for those of you that don't know about Pascal, I'm sure that I can refer you to anyone of our University colleagues for information about it. But it happens to be a particularly well organised language for certain types of work.

The question as to whether the Department of Defence does or does not standardise on Iron man, on the implementations of Iron man that have now been begun is of course, still open to some extent. But I am credibly informed that if they do, then the EEC will probably swing in behind it and certainly the NATO complex will and the result of this will be to put great pressure on the manufacturers to provide suitable compilers for it. If this is the case then quite clearly it is a language to be reckoned with. The introduction of Cobol, I feel, had two important features. One of them was at that time business was looking for a language in which to express its problems. Secondly it was enforced by the Government buying agencies in the States. These two produced very important effects. I think Iron man could have an equally important effect and I think it's important that everybody here should be aware of this and that some people within the Universities at any rate should be studying closely what is going on.

Of course one would like to see more developments towards problem oriented languages and particularly those which will help the systems analyst. At the moment, we are constrained by numbers on the systems analysis side in handling things and I'm sure you will have heard from Hansen and some of my other colleagues about the difficulties that are encountered in carrying through this and the difficulties we have had in training enough people. But of course it should be possible to

mechanise some of the aspects of this and at least to offer some help to the analyst in carrying out his work. At my school we have a project called LEGOL which is aimed at analysing legal text, particularly statute law and the object of this is to try, as far as possible, to aid the draughtsman to get a decent draft which doesn't have two clauses which clash and doesn't have gaps where certain cases are just not dealt with at all. Legal people, in general, are very concise and they think clearly. The laws come out from the draughtsman and hopefully with very few of these in, but there are a few. Most of us who have fiddled with the tax system are only too well aware of the problems it creates. So there is some hope along these lines, along picking out a particular area of endeavour like that, where we might develop something that would be valuable to the analyst and which, perhaps will lead to more general ideas on how to carry out analysis effectively with the aid of the computer itself and I believe these to be important directions.

I want to come now to one of the things which we have spent quite a lot of time on over the conference, and rightly I feel, which is the communication between man and man. We have had problems with this. Even in the conference we have had this. When Sidney started off the conference and he talked about computer science I saw a number of members of the conference going smartly into a little shell which said "my God don't talk about that". Because their concept of computer science, I'm quite sure, consists of only writing compilers or writing operating systems. Now, of course, I could defend doing that as an intellectual exercise without difficulty. Anyone who has had to write an operating system knows a great deal about resource allocation. He also knows quite a lot about various interesting things, like how people behave when they program and what sorts of things he's got to look out for in terms of the load that will come to him from a general bureau environment from University operation and so forth. Indeed some of the things that one teaches are exactly this sort of thing. If he

can write a compiler he's a pretty good programmer because he has to do things in a regular and effective way and we have in fact researched quite a lot of this. It may not be realised altogether how well we've done that. Back in the early 1960's, if you wrote a Fortran compiler for example, it was reckoned of being an effort of 2-3 man years. Nowadays, I would expect a Fortran compiler to be a matter of man months, possibly only two or three, depending on how well the chap has been trained. So we have made some quite significant advances in the effectiveness with which we understand some of our basic ideas.

I think, myself, that some part of the worry about this educational aspect of things springs from misunderstandings about what computer scientists think is computer science. Owen Hanson declared himself, as not being a computer scientist, so I can leave him out. On the other hand, I certainly am, and have been for many years, but I don't necessarily believe that computer science ends at programming, and I'm quite sure Professor Michaelson doesn't either. The important aspects of computing which have to be studied nowadays relate much more to the introduction of computing into businesses and the resulting organisational changes which spreads over into systems analysis and operational research. It's very difficult to say what you mean by systems analysis. A system is anything that you cut off into some sort of bleeding chunk which is studiable. The primary problem that you are faced with is how to carve up the tasks that you have into sub systems that can be studied and dealt with effectively and in what way you can go through fairly rigorous procedures for producing what it is you need to fit these systems. And these are as much a part of computer science as programming is, perhaps more so.

Equally, one can spread over into areas which are essentially industrial relations, we teach our people a good deal of organisational theory which is taught primarily by the Industrial Relations Department. I see nothing peculiar about this, and I believe that some computer scientists, at any rate, ought to be studying it. If one goes back to the sort of matrix man which

Nelson wanted us to have, he didn't really say within what time scale he was expecting this matrix to be filled out, and I feel that he probably didn't mean that we should bring them, all matrix ready from the oven in the university, and in fact it would be impossible to do so. One of the problems that I think is not always faced is the fact that to talk to somebody about an organisation without their having had some experience of being in an organisation, is usually a fairly useless operation. I once addressed a group of young programmers and others on the subject of organisational problems and I explained how they might find a little difficulty in moving from one department to another and then getting information out of certain people within the departments. Moreover, they might find that they had to go very high in the political chain before this situation was altered, if it ever was. Now everyone who has worked in an organisation knows this to be true. He may regret it, he may even feel that it shouldn't have happened, but the fact is it does happen. It's very difficult to explain to naive under-graduates who believe that everything should be as it is in the University, where of course, we never have things of this kind, and he thinks that it really shouldn't be like that. Explaining to him that it is maybe unfortunate but it's the way the world is made up. We have to live with what we have not, with what we would like to have.

So people have to be trained in this area and I guess that Dan Cougar wants us to train them in this area, but almost independent of computer science. I find this, myself unacceptable. I believe, in fact, computer science, the things that you can teach in terms of what is sometimes called conventional computer science, and I will take out of this the mathematical element, where I require somebody to have prerequisite of high mathematics, and say that most of the things that relate to programming do not, in fact, require high level mathematics they require some understanding of figures, numbering, it would be very difficult to do sorting unless you at least understood

something about numbers. But you can do the examples with letters just as easily as you can with numbers so it's not that much of a problem to present it slightly differently. Now, much of what can be taught in these areas can be done either in the context of mathematics under science, or in the context of business, or in the context of medicine, or in the context indeed of almost any other faculty. I don't believe that it is necessary to have little groups within each faculty to do the same thing over and over again with no contact with one another and if that was what was meant by university activity I should be very upset. Universities ought to be as they say, Universities, and there ought to be cross disciplinary arrangements and there ought to be people working across boundaries. There should not be little hard groups. So that in fact, what I hope to see and what I would like to see, within Universities would be a core of computer science was developed which could be taught within a faculty framework in many places. What we really need, I feel, to go towards Wayne's perfect matrix man, is a series of people who have parts of the matrix when they come out of the University, but they do have a core part which is labelled computer science and in which they understand a great deal about computing. Not everything, we are never going to understand everything about it. Right now I certainly don't, I've been at it 28 years. We're never going to see the whole picture as a whole, but at least we can try to give our graduates some idea of what it is all about and what it actually means within suitable framework. I do not see the framework being necessarily of itself being attached to mathematics or science alone. I feel it should be attached to a number of other subjects and as these subjects develop so perhaps the numbers will get more.

I feel that we've fallen into the trap if you like, of saying that because we want a lot of General Practitioners we should never train a Specialist and that because the proper training for a General Practitioner, is in effect, a training in Pathology alone over here and something else over there, we should never get a common core of medicine into them. Now this

would be silly, everybody goes through the same training, Specialist and Generalist. It seems to me the same general principals ought to apply in computing. You go through a certain base training and you marry up with various disciplines. This is what we have tried to do within the school, although we are, of course, not a multi disciplinary school, we are almost a single faculty of social science. But it doesn't seem to me that it would make any difference. So consequently, I would encourage people to do as much mathematics as they can. But there is a limit to everybody, or what they want to do about that.

Well, enough about the educational side for the moment. I am offering, if you like, a compromise point that somewhere between the point of view that says what we actually need is business graduates who have a smattering of computing, and those who say we should have deeply dedicated computer scientists entirely suited for systems programming and then try to teach them a little bit about something else on the side, or leave them because of their natural great ability to find their own way to other subjects. Well we've had a lot of that and not altogether successfully. But offering some compromise path and I think perhaps that's what we should see.

Because I think if one wants a justification essentially for increased effort in the Universities you've got to look at it from the point of view that there are people and I've heard their voices at this Conference, saying that they need more people in the computing business because at the moment there is too much mobility, well I wasn't quite sure what they meant by mobility and I have always associated lack of mobility with death. I have to admit that over much mobility is perhaps undesirable also. If you want to reduce mobility you have got to increase the supply, and if you want to increase the supply you've got to increase the amount of teaching and the amount of support given to computing in the system as a whole, and if you are now short of people then you should be planning to get rid of the shortage. The only way you'll plan to get rid of the

shortage is increase the amount of funds and effort in the Universities and I think I thoroughly agreed with a number of points made by Professor Michaelson, that on the whole this is not as well attended to here as it should be. I feel that I should support my colleagues in the University here in feeling that some funds are required more than are currently planned, if one can put it that way.

The mechanism I don't wish to comment on greatly, I have a general feeling that the computer board has served us well in the U.K. by syphoning off the funds before they ever got into the hands of the UGC. Essentially, ear marking them for specific purposes. Whether such a board would be appropriate or would serve you well is very hard to say. One has to staff such things and to a large extent the effectiveness depends on who you staff it with.

The question of post graduate training however, is I think, an extremely important one. I do believe that we are now entering an age where most people will do more than one job throughout their lives, possibly two or three. During the course of this there will be a retraining need. We have been told, and some of our union friends back home are telling us, that there will be very substantial unemployment. It's clear that introducing word processors into offices, the evidence is that we could replace one typist in four. Now, that's a fairly significant part of our work force. If we were in fact just to replace those and there were to be no reabsorption of the people we should have a problem on our hands in unemployment. I don't necessarily think we are going to have such a problem because I think there is an almost inexhaustible requirement for increased administrative staff to try and administer all the rest of us. We shall probably absorb into the structure, those people who are released by this kind of method. But nevertheless, it behoves us to take a look at ourselves and take a look at the whole structure of things. I think that we are going to get into more problems of retraining and this will be a question really

of post graduate and post experience training. I think we are not very well geared up to handle this at the moment either in the Universities, although they are probably reasonably well geared to it, but I don't think we are particularly well geared to it in business. I think there is a general reluctance to release people after two or three years to go and do a Masters Degree. There is a general reluctance at the later stage to co-operate with the Universities in producing doctorates. Yet, I believe these to be things which should be done.

I think we have a valuable contribution back home in the open University, which is offering opportunities to people to move from one discipline to another, to train themselves in new things. I think this is a very important thing for the future and I feel that you should look at it very seriously from both points of view. I personally believe in the computing field, if you're going to go through a career path which essentially starts with a programmer or an operator even, you're going to take them from the University, take them out as a programmer, you are then wanting them to become closer to the analytical field, to do analysis, or you are going to put them into the systems programming side, but in effect there is a need, after a certain length of time, to come back and retrain in the later state of the art. We have a lot of people coming back for retraining now, but under more or less voluntary programs rather than sponsored programs, who essentially have never seen anything except a batch system and who find that in fact, working with on line and terminals and so forth, a very strange activity and very different from what they thought of before. Yet, these people are fully professional in other respects. Should we not in fact take consideration of what we are doing with that sort of thing. If you want to move from being a programmer to being an analyst you need new skills, you need skills in organisation, you need skills in operational research, which you possibly never got at an earlier stage. You probably need some retraining in psychology. I think we don't think about this, we tend to throw people onto this without much thought about what their

background is or even what they understand. I believe we are irresponsible as employers in doing this. But I understand the pressures as well as anyone else. I've been an employer for some years now. I don't wear this for nothing. It's because I'm director of a company which employs people. The fact of the matter is that you don't want your best people to go away because you're doing a job and you need them. But if you don't think ahead, as to what is going to happen, you will have consistent problems within your business.

If you take our Government - we have had a series of horrifying problems culminating. Well, the most recent one in peoples minds is our vehicle licencing registration system in Swansea, which was frankly, a disaster from a computer point of view as a system, not because the computers were wrong, not because the programming was wrong, but because the system was ill conceived and had not been properly researched before hand and the staffing was done by simply taking people out of an environment in the county staff and putting them into the central environment and expecting them somehow to rise to the problem of marrying together, records from hundreds of counties. It was an irresponsible act to have believed that one could do this successfully, but nevertheless, one expects that people will go ahead, hoping for the best in such circumstances. There was a previous, totally irresponsible act in an attempt to put our traffic control on a sensible footing. The irresponsibility in that case lay in putting it in the hands of an interdepartmental committee with no project leader. The inevitable ensued. One hundred and forty million pounds later we had a fourteen million pounds system. This is the kind of thing which I'm afraid is endemic and it's something which will go on until somebody takes some action to train the people who are going into the jobs. I believe that serious thought should be given in every major company and in the Government about the future systems they are going to

put in and about what kind of training people should have who are going to run them and who are going to construct them in the first place. I have been saying this to our own Government, not altogether with effect, I have to admit. But I shall say it again, because I believe it.

Now enough of the problems that we have there. I want to wind up with two things. One is our relationship with the general public. I think, quite accidentally, and in fact in the interests of shorthand and general conversation we've introduced a lot of jargon. You can't altogether help it. You are dealing in a closed field. You're dealing with people who have already got through a lot of experience and you tend to use it like a shorthand. We use it among ourselves as a shorthand and there's nothing particularly wrong with this. What I'm afraid is, we project this into the public sector where people only understand rather simple words. The result of this is to create barriers which we would be well advised not to have. What happens as a result of this? Well, first of all the public reacts in a very predictable way. It sees us as an elite, which almost certainly we are, but elite's are not popular. So perhaps it would be wise to disguise ourselves a little. It sees us also, I think, as a threat. Once you don't understand something, it threatens you, even if it doesn't. Whatever the nature of the threat, and we've had various things thrown up, like privacy and unemployment and so on, which are being thrown up still, as canards if you like, which worry the public, and they have reason to worry, for we are not very forthcoming about what our position is on it. The privacy issue, I think you have acted as a society very responsibly in dealing with these things. We believe in the U.K. we have also acted responsibly in this matter and I believe that the majority of computer people are not suspect in any way. However, I also believe that on the whole our civil service is not particularly suspect in feeling that - in fact it's introducing an atmosphere of oppression. It's really quite an open and sensible thing. On the other hand

it gets tenaciously attached to things like the Official Secrets Act in a way, which makes one suspect that they are actually trying to cover something up, and I think we may well be in exactly the same position relative to the public, but because we are not understandable, we are necessarily suspected of being inimical. Now, that's something which we have to get out of. We have to spend more time talking to people and explaining that we are not actually harmful. That is if we don't want the public ultimately to react by imposing on us restrictions we wouldn't wish to have. It's very desirable, if you are going to be restricted and you may feel that you are in one way or another, that the restrictions are of your own making and not somebody else's and this has always been very much at the heart of the move of the British Computer Society in reacting to places like the EEC and so on where we feel we would rather draw the rules than have somebody in Brussels impose them on us. I feel that in fact, in any case like this you should think about this problem and take action accordingly. It's a social problem, it's within your own power to act within your own social environment. But you should make your voices heard and in a clear and unmistakeable manner not in some jargon which nobody understands.

The final thing I want to say is something which I've been trying to say for a long time now, with very little success. In almost every other endeavour the users are the king of the marketplace, and yet I find that in computing, the users are divided, they do not understand one another. They do not even speak to one another about what they do. They group themselves around the manufacturer, a manufacturer. They talk, maybe among themselves, but it is implicit when they talk that they are only talking about that manufacturer's machinery. Now this is a very curious state of affairs. It's unlike any other endeavour that I know of. The users should, they have the power, to exert influence on telecommunications policy, on manufacturers policy, on educational policy, and yet they do none of these things. They take almost

no steps, almost arrogantly no steps to inform anyone of what it is they would like, interesting themselves in the policies themselves and what should be the policy for them. It has seemed to me absolutely undoubted for many years that it would be in the users interest to be able to change machines on a rolling replacement basis. This requires an act of standardisation if you like, and it requires the users to get together and agree that in fact they are going to put the manufacturers in their places. If there are a number of manufacturers representatives here, I realise that might not be wholly acceptable to them, but nevertheless, it has seemed to me crazy that users should put up with what they get, which essentially is, we are locked into so and so and when we actually come to it we find we have to pay enormous sums of money to get away from the situation. That's crazy, it really is.

Equally, we have heard fortunately at this conference, a certain amount of reaction to educational policy and that is good and I would welcome more of it. But it's usually in the total negative sense of saying we don't like what you're doing, we can't tell you what you ought to do, but nevertheless we just don't like it. Now this is also not good enough if I might say so. The University community is highly responsive and interested in what people feel and they are trying to do a job of producing people. They, of course, operate in a rather longer time scale than most businesses operate. It is not possible within my University to get a new course established in five minutes, probably more like 5 years, to get it properly established and before the first graduate is coming out. I'm going to take 3 years to bring the graduate out anyway. If I can get it through the committee's in less than two I'm doing quite well. Now, I have to think therefore in terms of what is going to happen five years from now, up to say ten years from now and you users can undoubtedly bring a lot of pressure to bear on how I'm thinking about that problem. We have been beginning to try and tap our user community and ask whether

it's useful for us to do courses on a terminal basis rather than on a one year basis to allow you to take nine years over a degree, quite possibly this is not a bad idea in some respects. It's more perhaps consistent with what the polytechnics are doing in some ways, than what Universities are doing. But there is no intrinsic reason why the University should resist the idea. But essentially we want to hear, is that a good thing, would you release people to do this? On the basis of one term every so often. For Masters Degrees would you release them for one term every year rather for 3 years. Rather than attempting to take somebody in for one year at a time. We've got some preconceived situations in the Universities, but these are based on the past and we're not inflexible, we would change. But we don't know whether we should. Because in the end, if we do it, and people come to it then we can regard it as a success, but no more than industry do we like embarking on a project where we can't see the outcome. Please, we would like to know what our market is.

These are things where I feel the user has fallen down and is not really giving of his best. I would like to feel that I can persuade you here, who are users and major users and who are influential in this thing, to spend some time thinking about it and, if you like, bring pressure to bear in the right quarters to get the right decisions made.

Well, I think this Conference has been an extremely interesting one. I've heard a lot of very interesting things. My visit to New Zealand has been super and everybody has given me a marvellous time, so I hope in bringing this together in terms of the different forms of communication that we have to deal with, I have done the task which you set me of winding the Conference up.

KEYNOTE ADDRESS

"WHAT IS TO BE DONE?"

by

PROFESSOR SIDNEY MICHAELSON

After graduating as Royal Scholar in the Mathematics Department at Imperial College, Professor Michaelson held senior lecturing and research assignments there, most significant of which was the design and construction of digital computers, particularly the Imperial College Relay Computer. This was achieved with a background of research into Numerical Analysis and its application to problems of physics and engineering, including the solution of ordinary and partial differential equations, evaluations of singular integrals, vibrations in journal bearings and the design of chemical plants.

Since his appointment as Director of the Computer Unit in 1963 and later as Professor of Computer Science at Edinburgh University, Professor Michaelson has been engaged in some most interesting research involving computers. His current activities include the use of computers in archaeology for identification of texts and the fitting together of fragmented objects. He is also researching the analysis and design of computing systems involving Queue-theoretic models, measurement of systems and the prediction of behaviour of systems.

Papers which he has authored span a variety of content including physics, mathematics, the Bible, and the ancient Greeks.

I should like to start by thanking you and the New Zealand Computer Society for giving me the opportunity to visit your delightful country, to enjoy your slightly excessive hospitality, and tell you what I think you ought to be doing; that's always a nice thing to do.

I'm very impressed by the size of the audience that the Society has brought together. It shows that computing in your country is healthy. In spite of the fact that many of you have told me very often, it's only a small country. You have a lot of computer people active enough to come to a conference. That's very good going - I hope you keep it up.

Now, in delivering the sermon, one really needs to take a text and the text I want to take is that "It's all up to you". You are the custodians of the society in which you live. You may well also be its destroyers, and it is that contradiction that I want to explore a little.

The sort of life your children, your grandchildren, your great-grandchildren will live depends very much upon the actions you take. Normally we think of human society as a self-healing organism - something that has a great deal of stability built into it. Something which reacts in the old engineering sense of Lenz' Law, to undo the action that you try and do to it. But it need not be entirely true. There are very many circumstances in which society, as we know it, could be effectively destroyed - not necessarily by killing the people. You don't have to have atomic bombs dropping all over the world, or bacterial warfare let loose to destroy society. All you have to do is to stop people communicating with one another. And computers are a very potent tool for that task - for the task of stopping people communicating. Human communication does not just depend upon writing things to one another, or upon speaking to one another. We communicate non-verbally a very great deal. Much of our communication is done unconsciously;

this is why businessmen like to meet one another instead of just doing things by letter. This is why they don't just do things by telephone. They have their business meals, their social occasions; they make deals on the golf course. It isn't just wilfulness, corruption or any of those things. It is that people communicate by a lot more ways than the verbal things we first think of. The use of computers, however, could inhibit this other form of communication, could apparently completely remove the need for it, and it is the sort of thing that could come about by accident. Many of the distressing side-effects that result from technological advances come about by accident. We're all honest men, we all want to do good things. We don't always manage it though.

The people who invented motor cars weren't trying to destroy city life as they know it, they were trying to enhance it. They didn't manage very well, but they tried. The city engineers who cut cities up by roads that one can't cross, who allowed vast tower blocks to be built in the most remarkably inconvenient places for the people who live in the neighbourhood - the people who zoned work places and living places to be now miles apart, so that everybody has to travel long distances to get to work - were trying to be helpful, were trying to improve the quality of life, for people to live in one district without the unpleasantnesses of work amongst them. But instead of that they have given them instead the unpleasantnesses of traffic jams. You don't know what traffic jams are; you are very fortunate. But coming from London, as I do, I have had a modicum of experience of traffic jams and I can assure you that I would much rather have factories, offices and living places jumbled up together than put up with the transport problems of London. But the people who separated work places and living places were trying to be helpful and in that, as in many other cases, the evil effects of what people do are unforeseen side effects. They are not the intended

effects. Now you can't guard against unforeseen side effects beforehand - you don't foresee them, at least not explicitly. All you can do is set up mechanisms which will deal with them in due course as and when necessary.

So what I want to do is to explain how it is I think that computer use can have unforeseen side effects on a large scale and explain what sort of mechanism I think you should be setting up to guard against those untoward side effects. I have already explained to a number of other audiences what I think they should have been doing to guard against untoward side effects, and I emphasize the "you" because I happen to be here and not somewhere else. We all have a responsibility in this respect. It's a matter for the citizens of the country. It's not a technical matter - it's a political matter. It arises from technological advance, however, and therefore the people who are involved in the technological advance have some extra measure of responsibility. At the very least, they have the responsibility for communicating the possibilities inherent in what they're doing to their fellow citizens. Well, what are the possibilities inherent in computer use? In the days when I started being involved with computers, they seemed fairly restricted. We invented computers to solve the numerical problems we were faced with in things like crystallography, molecular structure, engineering design, survey analysis, quantum-mechanics. One or two people, for example the people in Cambridge University, in the late 1940's perceived the potential of the computer as a tool in business, as a tool in management, most of us didn't. The cost of a computer was such that it didn't seem likely that they would ever be very widespread. It took several people several years to build each computer. We used to say it wasn't a thing like shelling peas. One didn't turn out computers the way one turned out motor cars. One felt a little like Chippendale, producing works of art, and no two were the same, of course, and no two ever worked the same way either. It didn't strike us at that time that they were really the seeds of a major social revolution. I'm

going to repeat for you something which you must have heard lots of times, but I think it's important enough to bear repetition. The work of Gothenburg and Caxton was a liberating work. Printing was an intellectual tool that helped to liberate mankind from the chains of ignorance. It also had an enormous effect on society; amongst other things it permitted newspapers, and it permitted people to read the Bible for themselves, and these were significant developments. It changed peoples' ideas.

The industrial revolution of the 18th Century which, by the way, wasn't as fast as the word revolution might make one think, also had an enormous effect upon our society. It hastened urbanisation. It changed most countries of the world from places where about 85 - 90% of the population were concerned with the production of food, to a world in which 5 and 10% (in some countries under 5%) of the population is concerned with the production of food, and there still are parts of the world where 90% of the population is concerned with food production. But the places that are more likely to be affected by computers in the near future are the ones in which a very small percentage is affected by food production. Now those two things together are comparable to the advent of the computer because the computer is an intellectual tool which also has very large industrial uses. This, it seems to me, is a very significant difference from the earlier causes of social change. It's going to affect everything we do, intellectual as well as manual and mechanical. We can't flee from one to take refuge in the other, because they're both going to change. We're going to have a more all round change than we've seen before, ever. That's the first thing.

The second thing is that the computer is a more comprehensive tool than anything we've had before. We know we can use computers in all sorts of ways. There's practically nothing we do that can't be modified by the use of a computer of some sort or another and it's happened at a time when the whole of our technology is advancing quite rapidly, with the

result that computers, which we originally thought of as relatively expensive and rare things, are now cheap - not quite as common as dirt but getting that way.

In 1962 it was authoritatively stated in Britain that one shift on one of Professor Kilverns' Atlas computers could do all the computing for all of the universities of Britain for the foreseeable future. Well, you know how right that is. But you may not be aware of the rate at which we are now distributing computer power. For example, any individual university department in Britain is now finding that it can buy a machine with the power of four Atlases within its budget. This is quite a considerable change, for one can buy for a few dollars, something under a hundred dollars, a computer with about the power of an Atlas, that sits on one's fingernail. That's a fairly considerable change.

The rate at which prices are dropping is such that the sort of machine that cost a hundred thousand dollars in 1968 costs about ten thousand dollars this year and can be expected to cost a thousand dollars in ten years' time, and so on, and the rate may well accelerate. You can now buy, if you buy in lots of say a hundred thousand, computers to use as components in building things, at about a dollar a time. This means that one thinks of them in a very different way. There will be computers in everything - chips with everything, as the phrase has it.

This is a very different state of affairs from things we have seen before. Never before has advanced technology been so freely available. Never before have people been able to make use, at their merest whim, of such advanced technology. For example, people have talked previously of advanced automation and have built automated factories of various sorts to make things like pistons and piston rings, machine blocks for engines, they've made transfer mechanisms and they've made things like automated chemical plants, the sort of thing that takes three thousand people to build,

then runs with thirty people. So that one can arouse very false hopes about the employment situation in a district, import a lot of strangers, upset the local society enormously and then effectively abandon it and leave it with nil employment, having completely disrupted things.

One can now think about automating things far more than one could before. If one is worried about reliability or robustness, one can use several mechanisms simultaneously; they're cheap, it doesn't matter. One can go in for one's favourite technology, brute force and damned ignorance, in the freedom of low cost. Old-fashioned sound engineering consists of brute force combined with ignorance, but ignorance that's known about. You know you don't know how to calculate the forces in a bridge properly, so you build the thing fifteen times as strong as you calculated. It doesn't fall down. If you use the best of modern technology and build the bridge only fifty percent stronger than you calculated, you can't even build it, because it falls down while you're trying to put it together. We've all seen this happen in the last few years. The same thing happens with aircraft. How many motor cars do people buy that have been very carefully designed and which need repair within the first couple of months of their existence? It doesn't matter where the thing has gone wrong. The point is that the process of production has been very carefully designed down to the last movement of the operative's fingertips and the process as a whole has gone wrong. I don't care whether it went wrong in someone's numerical calculation or somebody's estimate of how his work force would behave - the fact is that we've applied the best technology we have to sort out what's going to happen and we've got it wrong. And we continually do this. So there's a lot to be said for old-fashioned brute force. Work out what you need and put in a large safety factor. It'll go on working, it won't need maintenance, you can sell it all over

the world, an excellent way to do things if you can afford it and now we can. Now we can afford to put in two or three copies of a servo mechanism and do a majority vote, reducing the likelihood of a fault to the cube of what it was before, from one part in a million to one part in a trillion. You may as well neglect the probability of fault when you've triplicated something of that sort. We couldn't do this before. We can now, so people are going to automate on a larger scale because they will see the possibility of saving money that way. Large scale automation can, if it is not done carefully and thoughtfully, bring with it considerable social evils. In the United States they made an announcement a few months ago that more than fifty percent of the workforce was engaged in information shovelling. Now, when you bring in things like electronic mail, automatic letter writing, automatic letter reading, automatic letter answering, we suddenly put out of business something like 49 % out of that 51% I guess. What are you going to do with them? The object of bringing in the automation is to save money. If the professionals in the computer field do their job properly, bringing in the automation will save money. Perhaps I should say if they did it properly it would save money. If the professionals in the computer field let that happen they'll have a lot to answer for, won't they? We have enough trouble with our present levels of unemployment. Not all the people who are retrained will be needed for jobs connected with automation. Almost certainly the factories that produce automating equipment will be automated, won't need so many people and with a little care in our technical work you won't need many people for maintenance either, so what will you do with people? You can't afford to pay them at the rate you paid them when you employed them or you are not saving any money. So you are going to have to think very hard to think of new things for them to do, new public works, perhaps we will have enough teachers for a change. We've still got to pay them. So, we are in a situation where our work is capable of leading to social problems of a very large magnitude.

There are other aspects where the use of computers can upset our society. Problems of privacy, for example, are things which concern us all. How do we keep files private? Is it technically possible to keep files private? I don't believe it is. The best estimate I can make is that the only way to keep files private is to change the viewpoint of your society so that people do not tamper with files that they should not tamper with. It's a very difficult thing to do, but it seems to me that human ingenuity is such that any technical fix to ensure privacy will be broken one way or another. If the worst comes to the worst, you bribe the data processing Manager and get at the files that way. Right! You are unbribeable, but you don't handle all the files there are. You see, there are other people who aren't members of your noble society who don't subscribe to your ethical principles and these things do happen. People appear in the courts every so often on charges of this sort. They've been bribed to do something they shouldn't have done. So privacy is a social problem really. It's not really a technical problem.

The general property of computer use in society is that information is handled by machine. To handle information by machine we have to gather it first. Before we had computers, information gathering was a very difficult process. Very unreliable, because people didn't send you the results they promised to send you. You just couldn't do it other than by going out and getting the stuff yourself and your life was too short for you to collect much information. Now we're in the process of changing all that. We're automating everything. We're trying to produce a cashless society, for example. "We" is the large we, embracing everybody in the field in the world. I don't know if anybody in this particular audience is trying to do it, but I know people in the United States are trying to do it and I know there are people in Britain trying to do it, and a cashless

society means information about you flowing around the system. Every purchase you make, every journey you make, every time you pay for something, every time you book something, it's all flowing around as data in a system. It can all be gathered and collected, enormous amounts of data, but then we're inventing enormous files that could be used for this. We're producing more powerful machines that could be used to analyse the material found in these enormous files. And the banking files are big. The sort of files I'm talking about would, of course, be many times bigger. But a Government that was so despotic that it felt that it had to have the information about the populace in order to take decisions, could collect that information - the information will be there. The technology to store it will be there, the technology to process it will be there. The only thing stopping them doing it will be the social atmosphere in which they live, the social atmosphere you have created and the politics of the time. It will be possible for economically minded Governments, economically minded companies to replace human activity by machine activity in a wide range of fields. In such fields as teaching, it will be possible, under the guise of supplementing the teachers' activities to do better teaching, or under the guise of replacing the small number of teachers, who one can't for the moment afford to employ, it will be possible to introduce automated teaching. It will be done in some cases from the best of humanitarian motives to help the disadvantaged, because computers can give children individual attention that teachers in charge of 35 or 40 children can't do; because one wants to help the disadvantaged. But it is very likely that it will not stop there. An economically minded City Council, not your present Government of course, but some future Government that you have mistakenly voted into power, not realising how malevolent they were, could very easily take that to the next stage, of replacing all its teachers by machines and turn all the teaching over to rote learning or possibly something a little better. Centralised indoctrination would be at its

fingertips and children would never have to see a human being outside the family. They need not necessarily see one another for that matter. Their terminals need not be in the same room, the need not move from home. Of course, their homes needn't have any people in them either. It's not necessary to have people to cook. Machines can do the preparation automatically, cook it automatically, dish it up and destroy the dishes just as well as a human being. Paper making can be automated beyond human activity so that you make the dishes, use them with the intervention only of the human being eating the food. Your nurseries and hospitals already have automatic monitors so that the nurses are alerted when babies do things that aren't predicted. Why have nurses? If you produce babies fast enough you needn't bother to keep them alive if they don't do well and if you do keep them alive then you could tend them automatically. Automation can be much more capable than you think if it's driven by a computer because it can respond in much more delicate ways to what is going on about it.

No, we don't have intelligent machines, but we do have machines that are beginning to behave in ways that can be described as showing signs of intelligence. It's just a beginning. We have the first inklings of some things. We are beginning to see some of the ways in which knowledge stored inside a computer can be used by the computer to answer queries, can be used by the computer to make plans for solving problems. It takes a lot of computer, but all of a sudden we find ourselves in a state where computing is cheap and is going to be cheaper. One of the major obstacles to the brute force aspects of artificial intelligence has been the difficulty of providing enough computer power. We can now see the possibility that that difficulty will wither away. And under those circumstances we can expect even our present primitive artificial intelligence techniques to work relatively well. We can expect improved techniques on which

people are working very hard to work very well. Those of us who are engaged in such innocuous pursuits as designing multi-computer systems could very well be providing just the computing systems needed for very advanced automation, and the very advanced automation could be used to cocoon a human being from the cradle to the grave.

Well, maybe our children will want that sort of society. Maybe our grandchildren will want that sort of society. I don't know. But I do know that it shouldn't be put there for them without their conscious choice. I don't think that it will be put there by deliberate choice of the people who put it there either. I think that we will gradually drift into it in the course of making life better for some people, in the course of somebody or another making an honest profit. We will put together a monster that will devour our society and I am worried about it. I hope I have conveyed the grounds for my worry to some of you.

This is one of the most important reasons for the specialist in computing to communicate with the rest of society. To tell them about the things that computers can do that they would like to have and to make them see that to have the good things without some of the bad things they have to be aware of the possibilities. They have to be politically alive. They have to behave as educated responsible citizens. It is our duty to awaken our fellow citizens to what is possible. To point out that it hasn't happened, it is not at the moment happening, and if they keep an eye on things it need not happen at all, but that it does depend upon their awareness.

Now, how do we do that? What can we do to ensure that our children and our grandchildren live in an alert, politically aware society, that has sufficient technical understanding to keep an eye on what is being done for its benefit? I see no solution other than through the educational mechanism. Education is a thing of slow growth. The sort of technical change I am talking about will become possible within twenty

to twenty-five years. At the moment that seems like a long time and I shall probably be dead by then, and probably so will some of you. But in fact 25 years passes terribly quickly. When I look back and think that I was involved in building my first computer thirty years ago, its gone like a dream and there has been the most colossal change around me in that time. The same thing will happen in the next thirty years. All of a sudden you will look back and realise that it is thirty years since somebody said "Watch it Mate" and you won't have watched it and you will be in dead trouble.

Education changes at a slow rate, its a thing of slow growth. You plant a seed and you nurture it and it often dies. If you plant enough seed, and look after them carefully enough some of them survive and grow into reasonably healthy plants, but you have to do something about it. You've got to start - if you don't start you never get anything. It doesn't happen by accident. How are you going to start? My view is that you should start in your Universities. I am about to bite the hand that feeds me, with the greatest of pleasure. I have been to two universities while I have been here and I have talked to people from another two and I perceive that your Universities are not performing the function that, in my opinion, they should be performing. They are not educating their students about computers. A slight exaggeration; there is a small amount going on. But the amount that is going on is in no one way commensurate with the need. In a university of ten thousand people, you just need teaching of a lot of students. You don't have it. You do not have teaching of enough computer science to produce specialists. You do not have enough general education of people who are going to be managers, lawyers, doctors, journalists, professors, civil servants, ministers, town councillors. People who are going to be in positions where they advise or take decisions in fifteen to twenty five years time. You are going to have a whole generation grow up having missed out on education about computers. This is because you haven't got enough people dealing with this subject in your universities. The way to teach geographers

about computers is to have their geography teachers teach them about computers. And in a university the people doing the degree teaching of geography cannot be expected to teach their students about computers and their effect on communities unless they themselves use computers to get some feel for what computers are, how they behave, how people use them, unless they have some intuition about computers and computing.

There is nobody to teach them, there is nobody to help them use computers. So it seems to me that one of the things you need to do is to strengthen your computing centres. The people who do business studies need adequate teaching about computers, not only as tools in business and in management, not only to be told about big data bases, information retrieval, economics of computing, punch cards and all that jazz. They need telling about computers as they affect other people. They need to understand how the end user sees computers. They need to understand how computer use can affect the rest of society. And that means that they need talking to about this effect on the rest of Society. And you get to them, in many ways, but most easily in the universities. Anybody who is going to be in a position of responsibility in the middle of their working life, or earlier, in some cases, should have some understanding of computers. They are going to affect everybody in society in considerable ways, ways that we can't now predict. So we need to have around us people who will react to what they see happening. We can't tackle it as a shooting problem, old fashioned way you solved differential equations, starting at the beginning and working forward. We don't understand the dynamics of the situation. But we do know that within very wide limits, whatever the dynamics is like, if we insert a negative feedback loop, something that senses what is going on and delicately tries to slow down what it doesn't like, we stand a good chance of getting the sort of behaviour we want. And in a social

situation the feedback loop is provided by people with social awareness, people who have some quite minimal understanding of the thing that is affecting society and look for its effects and try and do something about it. Most of these people are taught by departments that have nothing to do with computer science.

So from this point of view, you have to strengthen parts of the universities that will affect the general teaching of the university. You can do it by strengthening the computer science departments, you can do it by strengthening departments of modern studies, if you have them in your universities, you can do it through the administration, you can do it through the computing centre. But some way or another you ought to do it. That's not enough however, you also need specialists, well educated specialists, who can play their role in spreading a moral understanding, an ethical understanding. People who will join your society, subscribe to its ethical principles and try and spread them. This means you must have reasonably well educated specialists. Of course you need them for much more down to earth and selfish reasons. It costs fantastically less to produce a system with four or five, highly competent, well educated specialists than it does with a thousand warm bodies, as all the big computer manufacturers have discovered to our cost. We could do with, you could do with, lots and lots more of well educated specialists for your own selfish reasons in terms of your businesses, in terms of your scientific calculations, in terms of your process control, in terms of your management information systems. You need them all around the country, not least of all in the civil service. You just do not have those brains in the country. It seems to me you have a great need now, and you will have a desperate need in fifteen to twenty years time, and you are not producing them.

I do not believe that you have a single computer science department that has enough computer science staff to run

a proper computer science degree course. Now Scotland is only a small country. True it is twice the size of New Zealand, it has roughly the same number of students so we could do a good deal better than we do at the moment in terms of attracting people to the universities. No, maybe we have more students - I guess we have about sixty thousand university students, but we don't have proportionately more students than you have and we could do a good deal better on that score. I hope we will. But we do have a couple of computer science departments that are sufficiently well staffed to provide computer science degrees that really teach about computer science. I would have thought that your country could reasonably be expected to have at least one such university. You haven't - you've gone part of the way and then the support has vanished. Now for the benefit of the reporters, let me say that this is not something that I cooked up with the people in the universities. This is something that has struck me as a blow when I went around the universities that I went to, and talked to the university people I have talked to. I have been very, very disturbed by seeing that you have not yet caught up anywhere near with your educational problem. You need to take it much more seriously. I have heard some people talking, effectively in the DP field, about how you need to put pressure on the universities to produce DP oriented people, to produce practically oriented people, and to a large extent you are right.

The essence of computer science is that in a sense the name is wrong. It is really an engineering subject in which people need to be trained and educated as engineers to do a job that produces something for somebody else to use. If you look at the OEC report, written by a committee a few years ago about the technology gap across the Atlantic between Europe and the USA you will find the committee that was dealing with computer science came to the conclusion that the gap was a significant one and that one aspect of it

was lack of theoreticians and that the sort of theoretician that was wanted was not the sort that would be produced by any existing university mechanism but the sort of theoretician who would appreciate that he was doing engineering theory. Trying to theorise about a thing that was going to do a job. That is what computer science is about - trying to bridge from a technology to a problem. To somebody else's problem. To a problem for which somebody else needs a solution. It is not the classical applied mathematics situation of a solution going around looking for a problem. It is an engineering situation. To teach engineering you have to teach what is a craft and an art. You teach people engineering in universities by disguising an apprenticeship. You are dressing up an apprenticeship as though it was a physical science. Well it has aspects of a physical science about it but they are not predominant. The predominant thing is design, and that is very much an art and a craft, and is taught by the students doing design side by side with more experienced designers who help them to get it right, and see what they did wrong the first time. This means that computer science departments that are going to teach properly, need to be staffed well enough to teach from a very practical point of view. They need to be provided with laboratories like engineering departments. You don't have one that is provided as well as that, as far as I know. My University does. It is one of the two or three in Great Britain that does. Scotland has done very well to set up for such a small part of the United Kingdom. In fact, it is not the only one. Strathclyde University is setting itself up in that way too. It too has seen the importance of regarding computer science as engineering discipline. This is another University in Glasgow, regarding computer science as engineering discipline and providing it with suitable laboratory equipment so that the students can use computers in realistic situations. So they can handle big files, so that they can be forced to worry about scheduling; so that they can get some understanding of what's involved in giving such protection as one can envisage; so that they can think seriously about the problems of doing computing at opposite ends of a long phone wire. All the

practical problems from which you suffer are things that the students should have the opportunity to develop some intuition about - at least in one university. You could reasonably decide that the others are going to treat it as a pure science, but you should at least have one that takes it seriously, and that has enough staff to do the teaching that is needed.

How much staff do you need to teach a degree course? Well it depends on whether you have a three or four year course for example. The Scottish universities have the old fashioned system of two years in which the students do fairly general studies and two specialist years. Our experience is that the specialist years alone, take up the full teaching efforts of six to eight members of the academic staff, plus a lot of technicians, plus some programmers, all working together very hard, to keep pace with the rapidly changing needs for systems that bear some relationship to what the students need to understand about real life. Working like the devil to keep pace with the practical work and the projects that the students have to do. So my guess is that if you have a 3 year degree, you will need 8 or 9 teaching staff. Now if you have a four year degree you will probably want 10 or a dozen, plus supporting technical staff, programmers, technicians plus laboratory equipment which unlike some universities must not be just linear amplifiers, but your actual computer.

How can you expect a student to understand when an operating system is about if he's never fiddled in the works of an operating system? And how can you expect anybody to do any computing if students are fiddling in the works of the operating system? You must obviously keep your service computing and your operating system teaching separated. And that means the computer science department must be provided with "hands on" facilities. OK the computing centre can run them if you can persuade them to - it is much nicer that way. They run them, you use them. But it is not so easy to persuade them to add that to their burdens, they've got enough trouble with computing as it is. But somewhere between the computing centre and the computing science

department you've got to provide enough "hands on" equipment for your engineering teaching.

Your students have to be in contact with what happens in the world outside the university. This doesn't mean they've got to go out for a sandwich - that's almost certainly wasting their time. No, you've got to bring the outside world into the cribbed and confined cabins of the university. You've got to bring people in from outside to talk to the students about the work they are actually doing in practice. And you've got to let the students try and do the same things for themselves inside the university in a teaching situation where you can keep an eye on the. It's no use sending them out to be the tea-boys or to hump print-out around - that's just a waste of time. And I've seen that happen in places. They've got to be taught during any period that they are doing practical work related to commercial and industrial computer practice. And you can't send people out to a company and expect the staff of a company to put the time into teaching the, that your teaching staff do.

So you have other requirements, there are particular qualities that are needed in the teaching staff - this is difficult - its hard to find them. We, my Department, has on occasion had posts vacant for a couple of years that we have filled with temporary appointments while we've looked for capable people - people capable of doing the particular job of teaching engineering that we wanted to do. We found them in the end - it just took time, patience, a certain amount of going to to conferences and persuading people that Edinburgh was a good place to come to. Heavens! You could persuade them New Zealand is a good place to come to - it is a good place to come to. I am very pleased to be here. But I should be much more pleased if I hadn't had such a shock when I looked at your universities.

I think you have not treated your universities well enough. The data processing community is a powerful voice in the land.

(and the Government, perhaps not with any great welcome, but it will) the Government will listen to your voice. It may not formally consult the NZ Computer Society, but it should do. It does, in fact in Britain, consult the British Computer Society formally to take its views about all sorts of things connected with Government computing and Government funding of computing. That should be happening here. If you raised your voices perhaps the Government would take some count of them. Certainly if the data processing community raises its voice about the need to fund certain aspects of work in the universities more seriously, Government might take notice. In the U.K. the universities welcomed the Government's aid. OK, we will treat computing as a special thing for a few years. We will prime the pump, but we expect you to divert some of your resources in support of other aspects of computing. We will give you money for equipment and support some of the service functions - you support the rest. You strengthen your computer science teaching. You strengthen the services. You provide all the departments in the universities, so that you can do something in the direction of general computer education. This was set by the Government, it was accepted by the universities and it worked moderately well. At an earlier stage the Government had said OK we'll provide computers for your central computing service - you can provide anything else you want to. And the universities took that one too, and the Government bought the computers and the universities staffed the computing centres and found they hadn't got enough money to staff them properly, and then had to do battle with the Government to get the centres properly staffed, and there we got support from the business community and the engineering community.

I would expect you people to support the universities in that too. Its for your benefit both in your working capacities and as citizens. You need an educated citizenry for the benefit of your businesses as well as for the survival

of the society within which you work. It can do nothing but good if the universities can teach teachers properly about computers, if the universities can teach civil servants and bankers and journalists and lawyers, and doctors properly about computers. And you should be supporting that. You should be calling for it. You should be pressing on the universities to start it. You should be pressing on the Government to help the universities start it. It needs pressure from inside the universities and from outside the universities, and you are the people who can provide the outside pressures. You know about the field of work. You know about the need for the universities to do it. You have the ear of Government through your businesses, through your business connections. When you say something, somebody in the Government pricks up an ear and worries. You should make use of that power for the benefit of yourselves, for the benefit of the country as a whole.

Well, I have preached you my sermon. Let me end with a quotation - a slight mis-quotation, in fact, from the League of Nations before the war. One of the Russian representatives there used to say something, which on mis-quotation becomes (I think it was Paplitvinov in fact) "The price of freedom is eternal vigilance. You must provide the vigilance somehow or other, or we will all suffer for it."

AN ABSTRACT OF THE

KEYNOTE ADDRESS

by

MR H. WAYNE NELSON

"NEW DIRECTIONS IN DATA PROCESSING"

Mr Nelson has been Manager, National Accounts Marketing, Burroughs Corporation, Detroit, Michigan, for the past three years.

For 12 years prior to his current marketing assignment, Mr Nelson was responsible for application software development for Burroughs computers. This included banking systems, industrial application, hospital systems, and scientific programming systems.

Prior to joining Burroughs, Mr Nelson spent nine years as a member of the research staff of the RAND Corporation, Santa Monica, California, where he was involved in research on the use of computers in military operations and the impact of the computers in social systems.

Mr Nelson has lectured throughout the world on utilisation of computers in management and logistical operations, and has attended our Conference courtesy of Burroughs Corporation.

As a private citizen today in the United States, I enjoy the benefits and services of a Social and Economic system which has become absolutely and totally dependent upon effective use of computing technology. Our energy conversion and distribution systems are controlled by computers. Our airlines spend more dollars per dollar of revenue than any other industry, so that they have the computing capacity to allow us to reserve seats, automobiles and hotel rooms. As a private citizen I enjoy a level of personal financial services provided to me by the thrift and the banking industry that can only be administered, and maintained, by a very substantial investment in computing technology, and as you are probably aware these industries today are significantly increasing the level of their investment in order to create the capacity for more of us to have more services, more hours of the day and night.

Over the past thirty years we have spent a tremendous amount of money and effort to establish this base of use of this technology and its advantages to our government and private sector operations. I find that there are a number of citizens and even executives and professionals, who look at this and say "Isn't that enough?" Surely can't somebody synthesize that 30 years of experience, and then isn't somebody smart enough to write a book to tell us how to do it profitably, effectively. I guess they all have a kind of a yearning, a hope that maybe there won't be the need for the pioneers of data processing tomorrow, who unfortunately in the past tended to end up with the arrows back here.

But if you have been through two or three generations of computing in your organisation or in your professional career you may have arrived at the same conclusion that we have, and that is that we have obsoleted the relevance of the last thirty years of experience. We believe that today there is a new plateau of systems design technology, we believe it is important to every user because its impact is



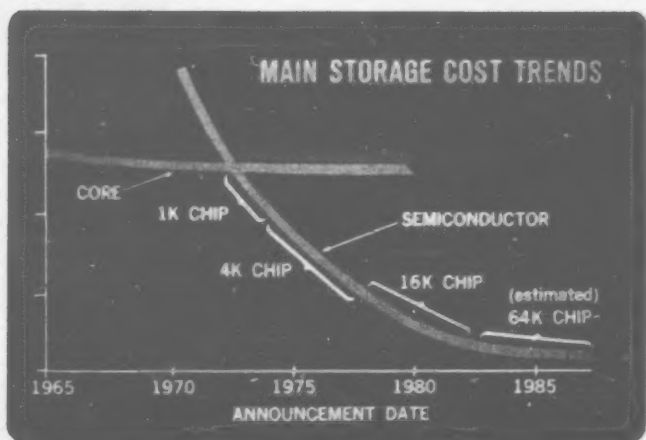
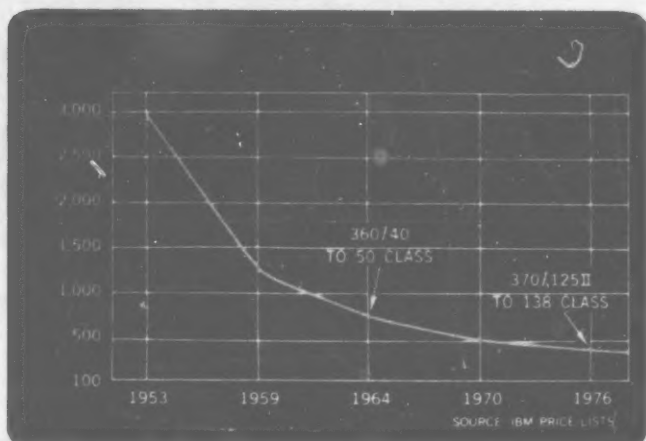
absolutely pervasive, it affects every element and aspect of the technology and its use. It affects how we design computers. It affects how people are designing their industrial systems, their logistics administrative systems and their management information systems. It affects the selection of tools and techniques that should be used in the design and the implementation, thus I argue it should affect the criteria that is used for the evaluation and selection of hardware and software to support business and government operations.

Now I am not talking about some concept that is off in the future on the horizon, that you should consider and evaluate because it is coming. There have been some ideas like that, that never did emerge in data processing. I'm, on the other hand, talking about a capability that exists. It has, in fact, matured through an evolutionary process that has been going on for well over a decade. That as a result it can be defined, it can be demonstrated, and more than that, today there is a body of third party evidence. There is a base of experience throughout the world by users in different industries and different classes of application with measured results. And those measurements are very important to us, because they indicate that the competence to use this new technology will, in the long run, reduce the relative cost of using data processing to support business and government operations, will significantly increase the effectiveness with which these resources will support those operations and for some users this new technology begins for the first time to offer an insurance plan. That is, it begins to provide assurance for the integrity for the much larger investment that any user must make in the design of his business systems, in the cost of the development and support and in the training and management efforts that is necessary to assure integrity in those operations.

But the measurements that we have, indicate to us that there is another side to the coin, and that is, we have evidence

that those people in our organisations, those members of our technical professional staff, by and large who have proved their value to us over the past decade by their ability to design, develop and implement our current systems, are not on the basis of that training and experience competent to deal with this new technology. Now I offer that this poses a very important problem, and I consider it first of all a management problem, because I believe that management itself must participate in some kind of a learning process in order that they can gain an understanding of the structure of this new technology, in order that they can gain an appreciation for its utility, so that they can interpret its relevance to their goals and objectives, and then if it is consistent, if it is reinforcing their objectives, they then must take the second step, they must invest and manage a process of training and of experience at the technical staff level, that I believe is essential in order to create the technical competence to interpret the real value of this new technology in the design of their long range business plans, and business data processing systems. Long range planning is the key, in my judgement and experience, to seeing the advantages of assimilating this new technology. Now I would suspect that all of you take that for granted. It is obvious to you probably that we must have this long range planning in order to be effective in achieving our goals. But the truth is that throughout the world many, if not a majority of using organisations, have data processing plans that are about at the end of their nose, and we have proved that this is not satisfactory, that we must have long range business plans complemented by a data processing plan to provide the context within which the potential advantages of this technology emerge and the various objectives for its use can be defined.

It is my belief that this new technology, as I term it, is based upon fundamental changes that have occurred in three major sectors. Hardware and the resulting economics of it. Communications, facilities and features that give us an entirely new degree of freedom in the instrumentation of our



social and business processes and Software, or at least as I call it, software today and its impact on system design on the productivity of development staffs and on the integrity on the investment that is made in those systems.

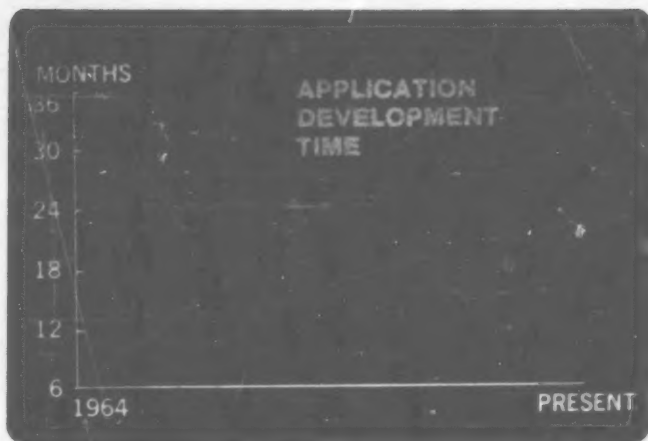
I would like to very quickly review each of these sectors with you, in order to tell you what I think is important to you, or why I believe it is important and to point out some of the directions of future development that I believe that we must also plan for. I am going to use some slides for this and I am going to look at some factors which are known to all of us, certainly to the professionals in this room, but I believe that they provide a context which I believe I can identify some of the issues that result from them that I believe are critical in their continued success.

Certainly the first well known fact that over the past two decades there has been a significant reduction in the cost per unit computation. The change in the economics of this form of energy is greater than the change in economics of any other form of energy available to man over the same period of time. It is my opinion that initially we were able to achieve this reduction in the cost per unit computation by building bigger computers. We built a big computer and then we built a bigger computer. But certainly, as you are aware, over the past decade at least, we have been able to continue this change in the economics of computers by building not only bigger computers, and we still need some bigger ones, but also by building today smaller increments of computing capacity. Now you can say, Wayne is obviously talking about the mini computer. Yes I am, in regard to its circuitry and its packaging and the resulting economics of that, but I am talking about our ability to create a comprehensive business data processing capability in a small increment, and I will try to differentiate that as I go through my presentation for you.

Today certain characteristics of our product line, and typically of our industry, if you will, is a range of computing

power. A spectrum of computing capability in increments that can be evaluated and implemented greater than we have ever had before. And the result of that, in my judgement, has been three major new directions and with each of those some significant new issues that must be dealt with. The first new direction, was in my judgement, a fundamental change in the mission of computers in business and government.

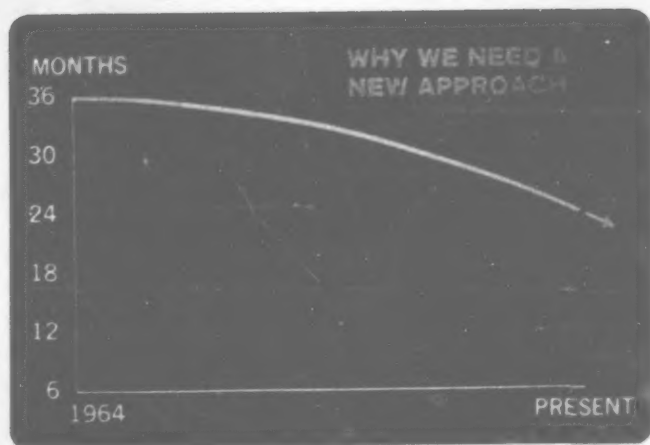
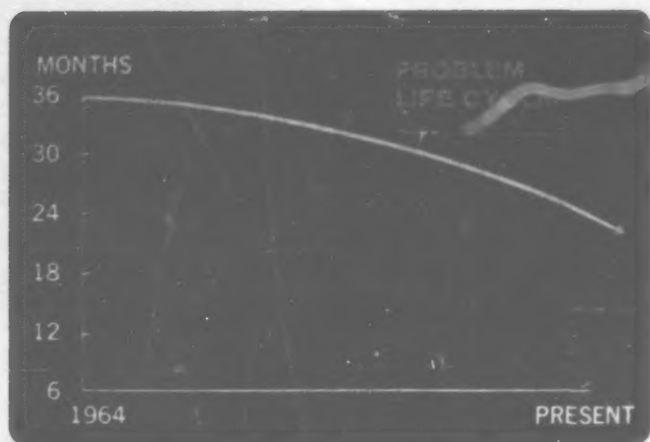
Initially, when we brought the computer from the laboratory we put it in the accounting functions of the business and government and we used it really to help us use our hands, not our heads. We called it data processing. We have eighteen years, at least, of experience in this class of application. We know indeed that the return on investment that we are after in business and government in this class of application is economy by increasing efficiency through the automation of routine administrative and logistical processes that generally deal with establishing physical control over the assets of the firm; fiduciary control or a more responsive and comprehensive customer service interface. But with this growing development of the technology and the changing economics of it we then saw a fundamental change in mission in the public and the private sector. If you will recall, the original stimulus for development of the computer, certainly in the United States, was the need for a computer capability that would allow us to improve the precision and accuracy of the World War I Army ballistics tables for Europe and the ability to solve some energy conversion equations for the Manhattan project. That was the core fundamental economic stimulus for this development. Then we began to change direction and we changed our direction to the use of the technology in the control of physical processes. In the USA we call this process control. We then moved into the accounting functions of business and government, as I have just defined in a class of application called data processing and now not exclusively, not entirely, but in my judgement, a major shift in mission that has resulted from the change in technology and the economics of it into these more fundamental issues of business operation, resource management and system management functions. And



as a result, if you will, from a professional standpoint, they have taken all the fun out of data processing, because given that change in mission, now here is the consequence. You bet your business on it, you bet your business on your competence in this technology. As a matter of fact, I think it is realistic to assert that any relative sized business or government operation in the United States, or Western Europe for that matter, has arrived at the conclusion that they are now as dependent on the competence to effectively use computers as they are dependent upon competence in whatever other disciplines are associated with the nature of their business. Because of that, obviously concern for computing has moved into the executive office and it has moved there for some important reasons.

First of all, management's question today, and a growing question is "are we able to be responsive to the changes that are taking place in the nature of their business?" So that as we move the technology from an ancilliary accounting service into the fundamental processes of the business, where in many industries the computer application is the business, there is now question of whether we can respond to the business decisions and the changes in those decisions that are necessary, and there is reason for this question. Let's look at a fact that we have in this regard relating to our ability to respond.

Back here in the early 1960's when we were doing those basic accounting applications, the response function was generally in the terms of six to nine months. Now there may have been an application that took a little longer but the consensus is that we were talking about six months for design and development of an inventory accounting or of a general ledger, or a payroll application of that nature. Now, that is no longer so, given the complexity of the technology we are working with as well as the complexity of the business processes we are dealing with, we now know that the response curve is over here some place. There is general consensus now that we are talking about 24 month's slide, that's the kind of frame of reference that you have when you



talk about the time to design and develop and implement one of these new mission oriented applications.

Well now, that may be not so difficult, if one could plan for it. But let's put that in the context of another fact that we know, and it relates to the life cycle of the solution we have implemented. Back in the 1960's when we were doing those kind of data processing applications, in general we have an idea that there was a consensus that the life cycle of that solution, the life expectancy of that solution, was somewhere in the order of three years. Many organisations, as you are aware, even capitalised this investment and amortised it over that kind of a period or a longer period. But we know that's no longer true now as we begin to deal with these fundamental issues in the physical process and the logistics of the business, there is a general consensus that we are now in the order of 24 to 30 months as the expected life cycle of that business system itself. In light of the continued technological changes that affect other parameters of business operation, or the competitive pressures itself. Well it is not hard to say "where are we going?" because we are there. X marks the spot when you can't get from here to there and it isn't something to speculate on, it has occurred. We have a set of examples where it has taken us longer to design, develop and implement a major business or social service, or government servicing system than the solution that we were implementing is relevant to the business. Now that is scary, and that is not a secret to executives throughout the world. They are beginning to be very aware of that and I could argue that we have little hope of beginning to push this life cycle back up. If anything, our technology is helping to feed this rate of change and I think then what we have got to do is find a way to be more responsive. We have got to find a way if we are to continue to do the mission that potentially we have, to service business in a fundamental way.

Now there is another aspect of management's concern here. We find that indeed, in most instances, management is using the computer as the mechanism or as the catalyst for introducing change into the nature of his business, into his business operation. Now that creates an important new problem. Ten years ago we had computer applications that failed for technical reasons. Today we have multi million dollar computer application systems that are failing for no technical reason at all. They are failing because of our lack of understanding sufficiently, or totally, of the sector that we are now affecting and how you manage change. We know from studies that have gone on in other disciplines that an organisation really has two missions - "Maintenance and Adaptation." In the maintenance mission we are trying to do a given job well, we are emphasizing standard operating procedures, automating them for increased efficiency. We are designing control systems. And in fact, in most areas of government and business, this is the dominant mission, this is the mission around which most of the management rating systems and motivating systems have been established. And what we have learned is that a business organisation has homeostatic characteristics. A business is a homeostat. My body is a homeostat. It tries to maintain my body temperature at 98.6 and if it gets cold outside it works harder to maintain that temperature. If it gets hot outside it works harder. It has this characteristic of trying to maintain that position. We have proved that once constituted in general, in this mission of maintenance, an organisation has homeostatic tendencies. It rejects change, it fights against significant change in order to maintain its position.

We know that in the adaptation mission of management of any organisation you are trying to define the job to be done. Here we are trying to respond to environmental changes that affect it. We are interested in organisational innovation to perpetuate the business. The focus is on effectiveness. This is quite a different vector of interest in the organisation. So that in the mission of maintenance we are looking inside the box; in adaptation we are looking outside.

But now when we were introducing the computer initially and principally in this mission of maintenance, by and large while there are problems we understood them, definition of need and of implementation, it was a different order of magnitude than when we change the mission and begin to bring the computer into these areas which are more susceptible or which themselves are a mechanism for creating the ability of the business to adapt, to change the nature of the business. There are several models that have been developed which give us an understanding of this process of change and the understanding of this work, I think, is important because it shows us that in this process of change there are these functions - the necessity to un-freeze the organisation or to un-freeze that homeostatic characteristic of the organisation then change and then re-freeze. Now if we were not changing, all we would be affecting is change and its interface, but now we have got a broader spectrum of a phenomenon that we must deal with.

More work has gone on to elaborate those functions that are part of this process of managing change. And the point is this. Here are the technical issues. Here is where the technical staff have, and are competent to operate, but the data processing technician is not trained to understand this total process and he cannot do it, management must enter these functions of un-freeze and freeze. In that entry process we know that it is absolutely essential that you must build a felt need at the operational level. We must define a contract for change in order to accomplish it now. In some instances that contract for change is based upon personal confidence. That the operational elements involved have personal confidence in the individual who is the change-agent. In other cases it is based upon establishing professional credibility. That you have helped them. That you do things, and in many instances the first thing to gain credibility is to do some small things for the operating units which are visible and which gain confidence. Or an other way, in other instances you can establish political trust, if not, coercion.

Then there is this recognition and responding to resistance. There was a tendency - don't talk to that organisation because they don't like what you are doing, go around somebody else. We know that to gain success here you must bring out the objections in order to deal with them, or you cannot be successful. We also know how to frustrate one of these systems if it comes on it. Don't object to it - study it to death. And then there is the problem of allocating joint resources and joint responsibilities. In many instances in the U.S.A. we have great mobility of DP managers. Sometimes it is fun to see them. They make a move at a time that they have gone through unfreeze and technical change. They have not gone through re-freeze. Now you do a post installation audit. Let's say three years later and nobody in the organisation ever heard of that fantastic system that that guy's resume said that he designed, developed and implemented. And it's because then of that importance of re-freezing, that terminal sequence re-freezing, institutionalizing it, and that includes the value judgements, is the system complete. Who owns the system, and then have you meshed that system with the value and other motivational parameters that affect performance appraisal. That affect communication patterns. Does the organisation have knowledge to maintain the system. One of the best ways I have found in my dealings, in order to be successful in this re-freeze is within the organisation to find some bright young man on his way up, who is recognised by senior management as a comer, and then make him responsible for the ongoing success of that system. Now share that responsibility with him. Put it on his back to see that it is a success. But that action is essential and we don't think we have understood it enough. We know that the capacity of an organisation, we know that the capability of an organisation, is the product of a complex interaction between these components. We know that we cannot affect one to any significant degree without affecting the interfaces, we know that the whole process in this area is critical to success.

Clearly then, that people-computer relationship, as we move this mission responsibility into these fundamental areas and the knowledge that exists of this process of change, of

how you administer a process of change, is itself critical to success. You know one might go further, say if not today then ten years from now, there won't be a technical problem we haven't really solved, but then we will be dealing with these social issues. Change is a social process. Change is the product of management, if it is successful.

Now obviously then we see that we have got to find a way of reconciling what appears in many organisational contexts to be a dicotomy - we have the user and the data processing departments in most cases. They have distinct roles and distinct responsibilities. If anything, the customer or user tends to be reluctant to relinquish responsibility for even the mechanisms or technology that is used in his operation. He would use the computer uneasily and have superior knowledge of the business, in most instances. Where as the data processing technician, by and large, lives in a limited world from that perspective of the business operation, has an inadequate business knowledge but is the advocate of change. So that we have then tried to look at who is successful in managing this interface and we have begun to identify two different people if you will, the technician versus his implementors. We have studied successful implementations where the management guidance has been successful and we have found a great deal of difference between these two DP processing manager types. And generally it is not this difficult to categorise the individual by looking at his focus on some of these issues. The technicians focus of effort is generally on design. On the other hand, if you talk to the implementor, the successful implementor, then his focus is generally on this entry process, this evaluation of need and operationalising of the goals, this termination process. He is concerned with it, he is aware of it. In output the technician will tend to be interested, or to discuss the product, while the implementor talks about the services, the benefits of the services, to be rendered. The technician talks about the fusion of innovation, the implementor talks about management of change. The technician will talk about knowledge of techniques, the implementor will talk about diagnostic facilities and techniques. The

implementor or technician who will talk about analytical prescriptive methods of that decision making, while the implementor talks about descriptive techniques. We have seen this difference that we have been able to view in the successful manager, or who is a member of a management team and is successful in this implementation process. And so it is our judgement, what we need in the future is a "Matrix" man. We begin to need a matrix of skills to be successful. In the past we have seen the university system create the technocrat, the technical oriented person, or we have seen the business schools create the business management person. Now we are beginning to find schools adjusting themselves, and changing to create the matrix man. Either the business specialist, who is sufficiently aware of the structure and of the technical issues to interface to the technical subject, or the technician who has taken on this business management training to be able to link to business goals, objectives and processes. So our judgement is, this is the focus of success, this hybrid, this matrix man, whether he is self educated or whether he is trained in this direction.

As far as participating in the management of change the other thing that we realize that these complex systems do not change rapidly. In fact you cannot rape an organisation with a computer, or management scientist. You must take changes in steps, there are many forms of limitation on change in those structures, in addition to the resistances which themselves are part of this institutionalist characterisation or characteristic. The question is, what is the balance between adaptive processes and maintenance efficiency in that mission? It is clearly a matter of judgement. We have seen organisations whose total orientation is here and they have gone out of business as the world has passed them by. One might accuse the railroads of some form of this orientation. On the other hand, in what we call Silicone Valley, south of San Francisco, we know there are organisations whose total orientation is to adaptation without any deep concern for cost or efficiency and clearly we have seen them go out of

business as well. So this balance is an important judgement of management, consistent with the goals and objectives of their organisation and the requirements of their market place.

And there have been some failures we have seen. Research has been done by members of the faculty at Stanford Graduate School of Business, Stanford University. A very large set, and they have documented this research. Fifty percent of the failures were caused by frequent changes of objectives or specifications, which really turned out to be the lack of un-freezing the organisation and creating an apparent need and a motivation for a lack of communication to users and inexperienced management in the operational areas. 90% of it has nothing to do with the fundamental technical issues at all and we have seen more of this coming.

Of course the other aspect of management concern, in addition to responsiveness and in addition to success in implementing these systems, is the integrity of the systems themselves. And certainly there has been a great deal of publicity that has related to the misuse, to the abuse of computer systems. The startling thing here is:- (1) Most of the abused systems that we are aware of have been identified by accident, not by sound audit and control and in addition to that most of these incidents that have been documented, the loss of the organisation is an order of magnitude greater than the average loss in a bank robbery. So that we have some real concern here. It is of concern not only to the auditing profession but certainly senior management in the organisation is concerned with it. Government regulatory agencies, we have seen become concerned, and a great deal of activity now going on throughout the world in all of these sectors to establish standards and capabilities for audit control and access control.

The one thing that we have learned is, that you do not take one of those systems that was designed for batch processing in the back room of the past. You don't take that and just put it on-line, in line and real time in order to make it interface with data communication or to a data communication sub system and in real time to the operation, and then try to add those functions and features which give you audit

control, access control to assure integrity. We know that these are fundamental design parameters of the system. The thing that we have had trouble with is that most of our computer technicians, programmer, analyst in our organisations - they don't even have an understanding, in many instances, of the traditional methods of audit and control, let alone the new innovative techniques that should be considered. As a result of that, management has had to make sure that the scope of the design team is broadened to include those people whose training and experience gives them knowledge and insight of audit control and access controls and integrity in operation. And so in many cases in the U.S.A. today you find the internal auditor, as the staff function within the organisation, now given final responsibility for an application design from a stand point there is adequate access and audit control built into the system. As you are aware in the U.S.A. any public organisation must have independent external audit of its financial condition for public reporting. We have found that the cost for the external independent audit is itself a direction function to the extent in which the outside auditor has co-ordinated with the inside auditor and his influence in system design to assure the existence of suitable and sufficient control mechanisms and that's difficult because in our experience in the United States the outside external auditor and the internal auditor act as though they're competitors for a given budget, they're in many cases antagonistic towards one another. Management must make sure that they are talking to one another and that there is a cooperative effort in order to achieve this.

Well no wonder then that if you go around the world today and talk to an executive he will tell you he's got a tiger by the tail. Because he has indeed begun to understand his dependence on this technology. He is aware of the change in mission in many areas, he has seen it as infiltration of his organisation rather than as planned developments and relative costs, that is the cost relative to other operations, as he expands into these basic mission areas, is going up and must be controlled, is out of control in some instances.

Well, at first I tried to say, the change in the technology has brought about a fundamental change in mission and as relates to that change in mission there are some critical issues at executive management level. Those issues dealing with our ability to respond, those issues dealing with our success in implementing changes in this sector, our knowledge of how to do it, the processes of doing it and the integrity and security of these systems.

I want to quickly change direction and discuss a second vector which relates to strategy of use of computers in business and government, independent of the mission objective itself. During the 1950's and early 1960's, given the structure of the economics of data processing that is the absolute level of investment and the mix of resources that were required for this investment, the strategy for employing data processing in both business and government tended to be centralisation of data processing services. That is in fact, in most instances, had to consider centralisation of data processing service in order to get at the economics of scale that was necessary to provide cost effective services. Now as our social expectations increased and the volume of work increased, or as our company grew we had to add capacity to these systems in order to support that growth. And so there was a tendency to go from that computer to the next bigger computer and run out of gas and have to consider the next bigger computer. Now that's not to say there is not a class of application where in business and government where centralisation does not give us a better perspective of the problem and allow us to see and give a better solution that clearly exists - but set that class aside, I mean on par, in both the public and the private sector, the trend over the last decade has been centralisation based on the argument - economics of data processing, and those of us who went through that period of time know that while we had created services that are vital to our operations, that we face some very severe problems. In some business we would take a process, which I would call natural to the logistics of that business operation, and we'd cut it up in a whole bunch of little steps in order to centralise a couple

of steps in a process. The result was process fragmentation and the loss of management accountability in the important aspects of the business performance or business operation. In other industries or other applications, particularly in the logistics area, we would add steps to a process to burden it in order to facilitate this centralisation. I might also say this, it is at least my experience that in the graduate schools of business throughout the world they have never achieved an optimal solution to that question of centralisation versus decentralisation of the management of a business. On the other hand one can see practically that over the last decade there has been a trend in the style of management of business and government towards decentralisation. That is certainly in the United States and Western Europe there has been a tendency to establish smaller units of operational responsibility and profit accountability, and in the United States we're seeing this move into government operations now, the tendency to decentralise this process of management and providing the services.

Well as we go in that direction, in the style of management of our businesses, while going in the opposite direction of further levels of centralisation in providing data processing services we've created some very very severe social dislocations and you can hear those arguments in the Board room, you can hear them in line management, you can hear them in the technical area where they are designing systems to satisfy what they perceive as the organisation needs. Well the fact is this, that the structure of economics of data processing as you are aware, is absolutely turned upside down and more so today there is another strategy that is available to organisations if it is consistent with the style of management and with their goals and objectives and that new strategy is indeed decentralisation of data processing service departments and some organisations woke up in the early 1960's and they had lost control of operations and so many of them were forced to have centralisation to gain control. What I am arguing is, I believe that this technology

that has emerged will allow us to consider decentralisation with system control. But I believe that's the critical element in the evaluation of this opportunity, evaluation of this opportunity of decentralisation. I believe the technology exists to allow us to decentralise through a network of computing facilities within which we can maintain adequate system control.

Now of course, in the USA that is popularly called distributed processing. Not an objective for everyone, although even today that is being used as a cover-up for gross expansion of many data processing budgets. But underneath it there is a sound structure of ideas or issues that are to be considered. Our experience has said that distributed processing involves the implementation of intelligent programmable nodes, terminal computers of some type, in the operating areas. Our experience says this can increase precision, productivity, accuracy and data generation, acquisition, audit and communication. We believe that those nodes, those forward nodes should themselves be able to manage networks or terminal networks within that organisation. Therefore they must be capable of handling elements of the database or files. They will, we believe, communicate through a network, through a communication facility to some host services. That is, we believe the network will basically be a hierarchical structure, at least in the near term. Successes we have had, indicate that we have found hard dollar savings in several areas. Improved cost control from the user perspective, improved operational responsiveness again from the user perspective better control and audit, improved database integrity and improved customer services. But in addition to those hard dollar savings, as I call them, I have been very impressed by the fact that executive management sees a different value to this. Executive management that I have been dealing with see this as the opportunity to map data processing onto the organisational structure in a way that is both technically feasible and cost effective. That could not be done before, and to them this is important in reinforcing the style of

management of the organisation. So I guess I have been impressed with the amount of money that the executives of an organisation are willing to spend in order to reinforce and to establish this concept of style of management.

Now I tried to indicate what I believe is important as a result of a change in mission and change of strategy of use of computers and the considerations attendant to it.

Let me look at a third one, and for many organisations I would argue that this third issue overwhelms those other two. Because if you study the cost of data processing for the last 15 years in major national or multi national corporations, or major government operations. In many cases you find their problem was not to match capacity to need at a given point in time - the problem was to manage growth. "Management of Growth". We have not, as an industry, helped management in the past very much in this important responsibility. Let me illustrate that. Generally we had a management team standing at some point in time and with their organisational elements responsible for business planning, they would construct a plan of their business operations through the future. It always went up. And then they'd ask the question, given that business plan, how do I match that with computing capacity which is cost effective at any given time, but is also responsible to the growth needs? That's a question that is relevant in that context.

Well, let's go back, if you will, for a moment. Allow me to assert that these are the primary resources that constitute the capability of a computing system. Now traditionally a vendor would take some increment of each of these resources, some concept of a balance homogeneous set of resources and he would package them together. Right. And then he'd give it a number of some kind in today's technology. Let's say, he called this increment of capacity System A. Then the vendor generally had a larger increment of those resources packaged called System B. It cost more, but it did more. Now let's go back to that business manager for a minute. So

he stands at a point in time and he can say - in that vendor offering here is System A, it has capacity now that will allow me to implement some important new application and then I can forecast in my planning that I'm going to outlive it, or I'm going to grow out of its capacity. But that's alright, here's system B. So that when I grow out of System A, I can move that out and I can bring in System B, and I can continue to service my organisation's requirements. But something happened. It never happened that way did it? Never. Indeed, when you went to move System A out to move in System B there turned out to be a mountain there. A mountain called "Conversion". Because in fact, he found that the environment within which his application ran in System A was different, that the environment within which they ran in System B. Goodness me, in some vendors lines today, there's even a different operating system in System A than in System B. There have been some organisations who have been in a perpetual state of conversion for well over a decade. Not because they are overly enamoured with that pornography, but because that's the only way they can hope to continue to support the growth of their organisations. Now, in addition to that, when we looked at the resource requirements of some mix of work they never constituted some balanced set of computing resources. They were always schemed in one direction or another, and certainly the work load never grew in step functions like that, they always grow in one vector or another depending upon the changes in the mission orientation, or the nature of the business operation itself. And so if you went back, even though the business says, I have outgrown System A, I need System B in fact, there are some resources in excess in System A so that excess resources of some kind or another became the rule rather than the exception. So that growth planning and growth management will assert for many organisations, has been the critical issue over the past decade and has been the most expensive element of their data processing operation. And I assert to you today that the new technology that has emerged will break the back of that problem.

It is technically feasible today to build between the logical system called the application and the physical system called the machine, a virtual system. That virtual system can allow the vendor to introduce new technology as it emerges and is cost effective, and yet ensure the integrity of the user's investment in his business systems. In addition, it is possible, and it has been proven today that we can build modular structures of computing capability so that we can better match computing resources to the profile of need, in being more cost effective and better management of the growth function. I believe in this respect, this is one of the important aspects of this evolving technology and it has been proved today, and I think it is up to the user to look at his costs over the past five years, identify the relevance of this growth factor and make sure that the criteria for evaluation and selection of hardware and software is within that criteria. I must be able to gather you want to address yourself to the question, then how do you establish an increment of capacity which is cost effective today, that will allow you to absorb that ongoing stream of invention and innovation, which every authority in our industry says, is going to go on for at least 10 more years. We're not down here, we're still up here some place. And you've got to say, how can I do that without that expensive and operational disruption of conversion that has affected all of us in the past? The technology offers us that solution and I think that's important to mention.

Well let me move quickly to this area I said data communications. This second area. Why is it important? Well first of all, our research indicated that 50% of your data processing cost is outside the computer room. It is involved in data generation, acquisition, edit, audit and communication. Secondly, in that area of organisational operation, productivity has been flat for over a decade. But I'm not arguing the necessity to automate those functions in order to put people out of work, but we have realised we must increase the productivity in those functions in order that the system

we're developing can achieve its effectiveness. In addition, even in those areas in which we have centralisation as the strategy for servicing the organisation, we must find ways in which we can create a man-machine interface to that process. We must design man-machine systems. And the data communication area allows us to make this process visible throughout the spectrum of that business operation in ways that we could never do it - through hard copy reports or other reporting or printing techniques. Now, as you know, over the past decade there has been an explosion in the design of sensing equipment, terminals, terminal computers and communications protocols, that in fact have resulted in the inherent ability to reduce the cost of communications. I'm not just talking line cost. I'm talking communications in the organisational sense in the organisation of data sense. That is generation, acquisition, edit, audit and communications. But the important thing really is that organisations have perceived this and have proceeded with significant investments to create these man-machine, these teleprocessing systems but they have done it in a way that has

- (1) locked them in time and
- (2) the criteria of the evaluation and selection of the nodes of the network, or of the other devices, have not understood, or have not acknowledged, the value of man-machine engineering.

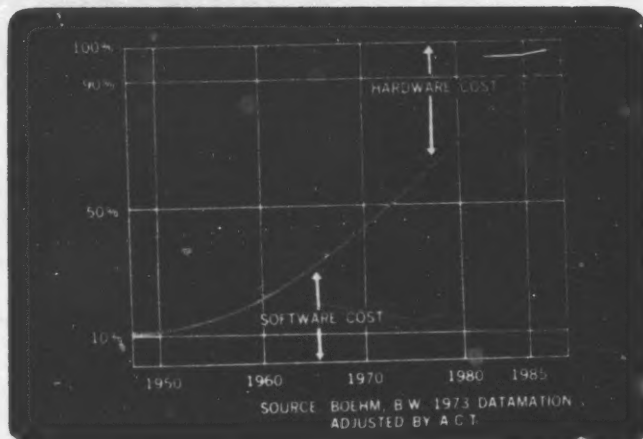
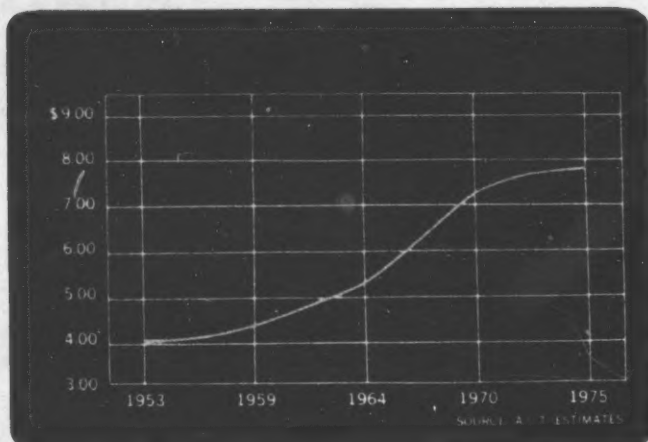
Now I won't over dwell on this but just as an example, in the terminal area we know that the ability of the terminal to have enough intelligence to be able to assist the user - the classical forms mode with flashing cursors or flashing fields with white on black, black on white or these things. We know that these are important in providing a better man-machine interface to that process. Further, we know the ability to design terminals in modular form, so that the input station can be built into the work station, so that the display can be put up on the wall or the counter to get it out of the movement of work. And put the power under the cabinet in order to protect it - of course it's shrinking, and it's going to go right inside there pretty soon. But that modularity is important. We have developed transaction

recording terminals and the importance of functional keys so that a standard application can look to an operator as though it is French, German or whatever jargon they speak, so that you can have standard design but man-machine interfaced in order to increase their effectiveness. I think that is important. Again the ability to have a dialography operator in order to assist, help, control and that has proved to improve the effectiveness of that man-machine interface. Above the terminal of course, is a whole host of nodes of a network we tend to call terminal computers, some people call them nodes per se. A wide range of capacity. But the important addition to that spectrum of capability is the ability of the terminal computer itself to manage files. Or elements of the database because we found outside the computer room generally, was a function called input control and when we studied the economics and the function of input control we obviously found that they were sending garbage back more than they were introducing status information into process of the business system. And so our experience was that even if you had central computing, if you could off load - i.e. if you could decentralise the job of data generation, acquisition, edit, audit, that you could improve productivity and that you could improve management control of the process and so many of the organisations in the United States who are proceeding with "Distributed Processing" even. The first step they take is to begin to implement data collection, edit or audit entry and similar functions in the forward operational areas, in order to put that input control function back within the organisation.

Of course, in a nationwide or multinational situation, the other node is a message switch, or what we call systems and communications processor. I think the important thing here is, in this area again is to say, as you examine any of the terminals, terminal computers or terminal software that is available to you today, you have got to know it's a snap shot in time. Indeed, if it's available in quantities that would allow you to implement a significant network of this kind, it's a snapshot of yesterday in terms of the engineering

technology and you've got to say, I insist that I be able to design my systems so that they are themselves independent of the terminals or the topology of the network, or the lines that they hook together. And the technology exists to do that today. And yet many organisations who were pioneers in this area find themselves rigidly locked to time because they set up criteria for the selection of equipment to build these networks, which today now results in the fact that they depend on a specific kind of transaction coming from one kind of a terminal, over one line fed by some increment of code in a partition of their computing capability and if they want to go back and add services to that location, or to that operational entity, too many times they have to go back and put another terminal hooked on another line serviced by another increment of code. The technology is beyond that, the technology exists to design your management systems independent of the specific elements within that network and I think that is an important aspect of the realisation of the value of this new technology.

Now I talk very briefly about cost of computing going down and cost of communications going down. Those are both headed in the right direction and obviously they are successful because we've built an industry on them. And they are essential but they are not sufficient, and you see the problem when you look at that other well known fact. While the cost of computing has been going down, and while the cost of communication has been going down, in fact the cost of using the computer in a business or government operation has been going up, and in many cases it has been going up at an increasing rate. Doctor Charles Lecht, who is President of the Advanced Computing Technology Organisation in the United States has very recently released a new book called the "Waves of Change". Some of you may have seen it. He gave me permission to refer, and to use some of the statistics that his research developed. Here's one. His research indicated that back in 1953 it cost us \$4.00 per line of code for our production systems. That is, take the



total cost of the development organisation divided by the lines of code that were actually resulted in the running code in the production library and he said that was \$4.00. He says in 1975 we were at \$7.80. Now, Professor Geoff Moore at Stanford Graduate School has done research exactly like this on a very broad base of samples, his cost functions are right on top of this, he says \$8.30. Now let's put that into the context of the changes in computing costs. And he shows this over that same period of time. Change in the cost of computing with this, the change in the cost of programming. Now the result is that in most areas of the world there has been a significant change in mix of resources to constitute an effective investment in computing technology. Back in this period of time we had a situation where generally for most of us, we were spending 65¢, 75¢ if not 80¢ of every dollar on hardware and then 15¢-25¢ of that investment went to those guys in sneakers and sweatshirts over in the corner who wrote code. And certainly that's changed now and we are in a situation where in our part of the world we are now spending maybe 25¢-30¢ for that hardware and we are spending 65¢-75¢ for that software development and maintenance.

Now I argue that the strategies which maximise the value of this investment mix are totally different than the strategies to maximise the effectiveness of this mix of resources. Further, Dr Lecht's reports, and several others that have been published recently, indicate to us that the cost of maintenance of these business systems is from 50 to 250 times the cost of the original design and development effort. Dr Lecht, I think, summarises fairly well the situation that many businesses now find themselves in. Let me quickly interpret the relevance of this chart. It says that we have approached a time here when characteristic of most of the businesses, 80¢ out of every dollar is in people and 8 out of every 10 of those people are maintaining old systems. Now that computes to say that to the business executive that 64% of his total capability is back there in the rear ends, not out here in the pits, helping him survive and be competitive. That's a serious problem. We had to begin to understand what was behind that. We certainly had to begin to direct our

111

research and our development throughout the industry in ways that would not only continue to reduce the cost of computing and communication, but would begin to bring down this total cost.

Our studies went on for a long time and finally we came to the conclusion that the basic problem began when we first built that computer in the laboratory. We built in a computing engine, and then we developed a traditional method of packaging that computing power which satisfied that engineering class of application. And then when we brought computing capability into business and government we brought the same computer, we brought the same people, and we brought the same traditional method of packaging the power of a computer. Now that was a serious error. It has cost us millions because of that and it has cost us millions to find an alternative to it. The characteristic of that process of packaging a computer is very very well, maybe painfully, well know to everyone here. It was called a program. The product was called a program. But let me identify the characteristics of that product in the business environment. A program in business and government became a specific solution to a specific problem. It was implemented at that very rudimentary level of abstraction. Second, we took all of the processes, logical or physical processes that were necessary to provide that specific solution to that specific problem and we integrated them together in order to maximise their internal efficiency.

Now, I argue that was correct originally. Because those original computers cost the most and their capacity was the scarce national resource. But we continued this when we brought this technology into business. Not only that, but we felt it was in our business interests to maximise the efficiency of the particular computer we had, so we implemented this process, integrated specific solution in machine dependent language. In assembly language at best in most cases. And you know, for 12 years I had that responsibility. My analyst programmer team had just come to me with a new application, It was just finished. Now

that meant that being finished, in most instances, meant that the user had thrown up his hands and said I'll accept whatever the heck you've got if it runs. In addition, it wasn't debugged, they didn't understand that, so most of those things had to be fumigated. But here they were, they had their new system, and by George, inevitably it was just then that the boss walked in the door and he'd say I'm sorry but there's been a change in our business. And having been there, I know there was a great desire to say "get the hell outa here". Because you knew if you played with that very much you had a very serious problem on your hands. It was our conclusion that that method of packaging the power of computing resulted in frozen applications.

Now we are at a point where there is consensus that change is the single most important characteristic of a business process, in either the private sector or public sector. That change is the characteristic that we must design to manage. We had a process here, a packaging technology that wasn't even sympathetic to change. In addition, as you are all well aware, the data that was controlled, and indeed was controlled in those specific solutions to specific problems, was not available in a realistic way to meet the other demands that management wanted. And indeed they'd come to us. Management would come to us, as technical professionals, and say "help me". Get me information that will allow me to judge the impact of a change that appears to be occurring in my business. And it would go from critical to totally irrelevant by the time we could write another program, because that's what we generally had to do and debug to get this report on their desks. We didn't help them and we didn't advance. And there then we tried the most disastrous of all, you know we said "oh if that doesn't help them let's inundate the so & so's with paper" and we created the report factory.

But we finally decided here was the problem. We decided we had to find a better way to package the capability of computing technology and then build computers to implement that strategy in order that we could be responsive and flexible. Now we went back to that thing called a program to get some guidance

and we found that 40-60% of the cost of design and development was involved in the implementation of those functions which quite obviously have absolutely nothing to do with running the business. Further, our evidence indicated that 40-60% of the cost of maintenance of that business system resulted in changes to those functional components because of changes in the computational environment and the organisational environment, if not the business environment itself. But you couldn't ignore them could you, because they're the interface that's where the application, that's where the business problem marries to the architecture of the computer and it's only if those are implemented effectively that you can gain the cost effectiveness of anybody's computer that you may have chosen to acquire.

Well, we said what do we do? The first assumption was let's break those apart. Wasn't it? And so the first milestone, I think, in trying to achieve, and trying to achieve flexibility, was this concept of functional specialisation. And so an organisation would take some people who wrote the best I/O code, they'd say "you go write it and put it in the library" and everybody copied it. And we began to get specialisation and our attempt was to have modularity in order to create the flexibility to absorb change. And I think the direction proved that the strategy was correct but as we proceeded, we began to learn. We began to learn that what we were really trying to do was to create what I call a data related service for that operational entity. We were really trying to implant those functions which dealt with data generation, data acquisition, data edit, audit, verification, communication, storage and retrieval, in order to make that primary resource called data visible to all of these individual applications and with the bias that's necessary for them to be cost effective, to make it visible to an entire new dimension of analysis and reporting. And then we said "let's implement them so that the user does not have to understand the technique of solution to have the utility of the service itself." And we've made tremendous progress in that direction. And I believe that this kind of typifies the current state of the art. Current state of

ENVIRONMENTAL SOFTWARE

- THE MANAGEMENT LINK BETWEEN THE COMPUTER, THE TERMINAL NETWORK, THE DATA BASE, AND THE INDIVIDUAL APPLICATION PROGRAMS

the art is a set of services to the application program which acts as an interface to the architecture and in fact create a virtual system for that application logic. What we are trying to do is to create a computational envelope in which that specific application logic can exist, but be totally independent of that environment, and yet we can service it with the effectiveness that is necessary to meet the challenges of the workloads that exist. Now we call that "Environmental Software" today. We think it is an important management link between the computer architecture, the terminal network, the database and the individual application programs. Somebody else in our industry, who has a tendency to establish de facto standards will some day find a new word then we'll all use that, right now I'll use the word Environmental Software.

It is important because it changed the level of responsibility between the vendor and the user. Before we sold you a computer and its operating system. We call ours a Master Control Program, and then together with whatever languages were available, it was the user's responsibility to implement all of these functions that were necessary to service business needs and control the flow of work through it. Today we move that from here to here. Today we realistically can say to you, you run your business and we will manage the computer interfaces and do it effectively, efficiently. And so there has emerged an entire new class of software. In my personal judgement it is as fundamental to the ongoing evolution of this technology as the design of the original operating system or compiler level languages. I think it is very fundamental. Then we have implemented this across the whole spectrum of our product line and I think we find that throughout many instances. Now that didn't come to us chiselled on tables, on tablets of some kind. Indeed in many other areas, functions similar to this are provided by vendors or by software houses. It's just that we believe that while modularity is important, while a problem oriented language interface is important, that it must be a homogeneous

set of services and I guess in that respect that we believe that the manufacturer is best able to do it because today we're talking about environmental software, tomorrow I'm going to talk about environmental firmware and the day after tomorrow I'm going to be talking about host services that maybe in micro processors, in the controllers or someplace else. I think it is a fundamental concept. It think it is a fundamental new development in this area and I have asserted to you initially that there is experience throughout the world that says you can improve response time. You can reduce the cost, in terms of time and dollars.

Another important thing it did however, it allowed us to marry those different classes of application. Let me quickly show you. Most of us came into this fascinating world of computers down here. This was this program integrated specific solution to specific problems. Right there. We proved that when you implemented one of those applications you really didn't help manage run the business. You didn't help management run the business. Now that was a disappointment to some people, but there's overwhelming evidence to support that so we defined another class of application in our attempt to instrument management so that they could better run the business. But this became the infamous M.I.S. goal of the late 1960's and early 1970's. We spent a tremendous amount of money throughout the world trying to implement M.I.S. systems and I think we conclusively proved you could not implement an effective M.I.S. system if you build it on that kind of a basket of eggs, or that primitive data processing technology. But those organisations who were able to separate data from the processors that use it, and prove they could maintain its integrity, they are now implementing these information systems, they are providing visibility of that information even throughout network structures.

Again database came into our technology with a myth. The myth was that everybody should try to build one totally integrated database with no redundancy and a lot of organisations foolishly spent millions trying to achieve that goal. We know

that is naive. I believe those organisations who are now working in this technology understand the goals to develop a modular structure of databases to increase reliability, increase security and give themselves the opportunity to decentralise elements of that database when it is technically and economically feasible, and within their goals and objectives.

Management science and operations research proved from the 1940's forward that we could build models of industrial systems, social systems, economic systems and if your technician is competent, if professionally competent, he can prove that his model is theoretically sound. The trouble is those models became gimmicks and playthings in most businesses because again, we couldn't instrument them with the dynamic changes that were occurring in the elements that were their sensitive parameters. But today those organisations who have achieved a structure, a technical structure, that represents a comprehensive data related service, they're now back implementing those resource planning and resource allocation and management models and doing it very effectively, and we are now beginning to see the true return on investment they hoped for.

I will close this way. I've said to you, for 20 years now there has been an evolutionary process going on, and at different points in time we created a new capability in computing, or in programming, and many users throughout this period of time, as they went from one step to another, have accepted certain of these pieces and they've improved their efficiency, their productivity and their effectiveness, but we believe today this has matured and crystallised and that in total today it represents a new structure for system design. We see new systems, new ways of structuring systems as a result of it. We see the need for large computers and as indicated we've just released a new one in the 50 million floating point operations per second class, because there are jobs that need that. But I do not see these large computers in the

future, in order to continue this process of centralisation where the argument for centralisation is an economic argument. Therefore, for most of us in the future, I see a multistage computing network to maintain control throughout that network. I think we will use this kind of environmental software I talked about. Our experience is that the control depends upon controlling the interface functions, and within that scope we can then have flexibility and access in processing of the data while we will still have batch jobs going on like we still have punch cards. We see that when those structures exist there's a tendency, more and more, to move towards transaction processing. Now in the United States a whole new sector has opened up. When first class mail went to 15¢ that was the breakpoint, and now the marginal cost of adding capacity to the network to handle administrative traffic and transmission of images is now cost effective and so anyone considering long range planning in data processing, the scope of their charter should be widened to include consideration of this kind of data processing within the organisational entity. I've tried to say to you I don't think you have to wait for anybody to develop some new theory or some advanced new technique. I really believe that what we need today is leadership. We need an understanding of the significance of this technology by management as well as consultants, as well as technical professionals, in order that we can give this guidance to management, in order that we can assist in the development or acquisition of this new and the meaningful value of this new technology. I guess there I'm pleased, because that means that those are people skills, not hardware, not machine skills that we are dependent on. I thank you most kindly for having invited me here and giving me this long to present my concepts and my perspective of this technology.

Thank you.

FORUM NOTES ON EDUCATION

6th Computer Conference,
AUCKLAND AUGUST 1978

The following is a summary of the forum discussion following the paper presentation "Teaching of Computing in N.Z. - The Dream and the Reality" by Dr Colin Boswell, M.Sc, Ph.D., MNCIS and Peter Melhuish, M.A. MBCS.

The forum was chaired by Dr Colin Boswell and this record was made by Jeff Cook; approximately 100 conferees attended.

In this summary, individual speakers are identified, when these were known to the recorder, or when they announced themselves. Since no audio recording was made, only an approximation to what these people actually said is presented here.

(Anon)

When we employ graduates we have to detrain them. They have been taught ALGOL and have been given a negative attitude to COBOL. The result is a strange ALGOL like way of writing COBOL which is unmaintainable.

(Brian Gibson, State Services Commission, Computer Services Division)

Our experience in a multi disciplinary shop is that COBOL people have trouble with algorithmic languages. We often recruit overseas.

COBOL programmers are obtained from Wellington Polytech and from CIT. We have people on the Boards of Control of the Technical Institutes and so have some control over the courses. We like an analyst to have a degree and to have practical experience.

Database techniques are of more use than File Design. Analysts, application programmers and systems programmers have different attributes and so used different training.

(John NINE, Victoria University)

Programming is easy to teach in a University.

Design is possible.

Analysis is almost impossible, only learn techniques.

We have tried using projects with 4 honours students but this is not possible with many undergraduates.

Notice that Dr Cougar said that Analysis had to be moved to the final year of degree courses in the States.

(Ian Mitchell, J.P. Scott & Associates)

Wherever a trainee comes from, we still have to teach fundamentals. e.g.

- data editing
- data structures for the particular machine
- data conversion
- program structure.

University students tend to be quicker to pick it up. You have to have someone to look after trainees, read their first code etc.

(Anon)

D.P. Projects have management problems.

In USA, MBA students are employed. These do courses in which the priorities are as follows:-

- 1st Management Techniques
- 2nd Project Management
- 3rd Business Functions
- 4th Programming & Systems.

(Anon)

University Students tend to be one-legged in Computer Science but they also need another leg of Accounting etc.

(Peter Melhuish, Massey University)

In N.Z. Universities, few students take more than 30% Computer Science in a first degree. The rest of the degree is formal from papers from Business Studies, Arts, Science etc. We have to consider at what point a person decides on the

career he will follow. Maybe he should have a first degree in something else (as is done in some overseas universities) then do Masters in Computing Subjects.

(Anon)

Every other profession accepts the need for practical training of the graduate, e.g. doctors and lawyers. Why should we be surprised if we need a professional apprenticeship?

(Margaret Fidow, Manukau Tech.)

High School Students are told by careers guidance counsellors and teachers that a high level of maths. is required for D.P. This isn't true and perhaps we should educate Careers Guidance people.

(Anon)

The problem is that students have a high level of expectation, caused by the syllabus and the year of settling in is really just the time required to disillusion them.

(Anon)

The problem is that Universities think that they must educate not train. They do not take responsibility for what happens to students when they come out.

(Mike Walls, Wattie Industries)

Worried about the growing obsolescence of his own training. Who will teach the non-new entrants to the industry? Is this a role for the universities.

(Ann Bigby, I.C.L.)

Said that she was surprised how little was new in this conference, most of what she had seen was taught on manufacturers' courses.

Suggested that sandwich courses of the UK type might be appropriate here to provide practical experience.

(Peter Melhuish, Massey University)

Massey tries to create industry-university dialogue.
Sandwich courses are over-rated but can soften the shock
in meeting the real world. Maybe they are more appropriate
for technical courses.

(Anon)

Management are Ignorant and Lazy!
They don't bother to find out what is taught.

(Anon)

Worried about preponderance of academics presenting papers
to the conference and a lack of reality.
Why is it impossible to train in 3 years in a university if
you can train in 3 years in industry?
Universities are arrogant in the choice of subjects, studies.
They don't care about the usefulness to the student,
industry or country.

(L.Groves, Massey University)

Students go to university without knowing what career they
want to follow. A suitable 3 year DP course would have no
takers (or maybe a few from industry) since school leavers
would not be ready to specialize.
We have a Diploma course for this purpose.
- for new entrants, retraining, updating and career advance
ment.
University degree courses must interest the students.
Courses prepare the students to go in many directions, not
just into industry.

(Anon)

Lack of communication between employers and the universities.

(Anon)

Communication skills are asked for, how can we teach them?

(Glyn Carpenter, A.T.I.)

Section 3.3 of the presented paper is disputed, in that

Polytechnic diplomas are of more than "marginal value". It has been found that A.T.I. Certificate holders are sought after and have had no difficulty finding jobs. This contrasts with 6th and 7th form school leavers who find it almost impossible to get a programming job. In addition, a survey of ex-students of ATI indicated that a high percentage stayed in DP and moreover stayed in DP in Auckland.

ATI is not bound by the NZCDP course as suggested by the paper, merely providing the opportunity to sit the NZCDP papers as a "backstop" for the less able students. We have attempted to widen the education given with Business Studies, Accounting and Communications English as well as practical programming and working experience away from the institute.

(Robin Stirling U.E.B.)

Maybe we should survey the expectations of students going into courses.

(Anon)

Technical institutes give an alround training, more in one year than in three years in university.

(Prof. Tate, Massey University)

Teachers Colleges are the only ones teaching communications skills and we could learn a lot from them. Teaching of English in Technical Institutes is only scratching the surface of the problem.

There are misconceptions on what universities do.

They are not educating for the DP industry alone.

Each course is a terminating course, at Massey there are 700 1st year, 200 2nd year, 100 3rd year and 10 4th year Computer Science students, i.e. for the most part the courses are service courses for other disciplines.

The 10 4th year students have high employability and we could sell 10 times as many.

(Bob Jones, I.D.P.E. Personnel Services)

We have difficulty placing degree and certificate holders. The low stream from technical institutes have trouble and often end up in operations. There is a neglected area - practical experience. There is resistance to new graduates.

(Anon)

With reference to the survey presented in the paper, this was based on DP Managers' opinions. Do they necessarily know what programmers do? What do programmers think they need to know?

(Anon)

DP Managers have a longer view of career development, promotional prospects etc, rather than job requirements now.

(Peter Melhuish, Massey University)

DP Managers were fairly close, often checked with programmers as they completed questionnaire.

(Anon)

Attitudes of graduates are more important than skills, they must be prepared to keep learning. We have a fast moving industry and we need to be self-educating.

(John Hine, Victoria University)

Universities can not cope with more students. Don't have the equipment, people and the techniques used are from the mid 60's e.g. don't even use the programmers work bench.

(Peter Melhuish, Massey University)

Is there a need for the MBA type of course or do we want to just produce highly trained graduates?

(Ian Mitchell, J.P. Scott & Associates)

Experience from the Auckland Branch program of courses shows there is a lot of ignorance about anything outside ones own job. He had to answer many questions as to what "word processing" is when a seminar was advertised.

Placing people in narrow jobs, requiring too low level skills leads to drop outs. e.g. three years as a COBOL programmer then drop out. Luckily this makes COBOL programming a suitable job for women as they usually get pregnant and drop out anyway.

School courses tend to be based on Applied Maths and tend to use BASIC if the school can afford its own computer. Some schools would like to use computers for accounting etc and for this the school mini will not be suitable.

Currently being discussed in the Auckland branch is a scheme where a company could adopt a school, bring in the kids, plan say a General Ledger, see programming and installation and talk with the accountants.

Personally he found that trying to return to university after a break required him to do prerequisite Maths papers before he could do some computing papers.

Universities could - waive prerequisites

schedule courses for people in industry.

Industry could - provide teachers, especially for case studies.

(Peter Melhuish; Massey University)

Massey post graduate diploma prerequisites could be waived.

The session concluded because of lack of time rather than lack of people wanting to express opinions.

No attempt was made to reach any formal recommendations as no real consensus of opinion seemed to emerge during the discussions.

FORUM NOTES ON NEXT CONFERENCE

6th Computer Conference,
AUCKLAND, AUGUST 1978

1. The Timetable, in broad outline, should be distributed earlier, ie. prior to arrival at the conference.
2. Timing of next conference - three years was considered to be too long in view of the rate of change and all present at the forum considered two years to be just right. It was queried whether the conference has to be held in the August University break and it was suggested that the May break would be just as good. It was also suggested that if there was no need for University Halls of residence it should be held at some other time eg June/July/September or October. A fairly strong body of opinion, was that if the N.Z.C.S. was doing its job for its members the next Conference would be in 1980. It was pointed out that N.Z.C.S. already had an agreement with A.C.S. for 1981.
3. Format - This conference was very successful due in large part to the exhibition. It was considered that the next conference may not be related so much to exhibition, but could perhaps be oriented towards extending more facilities to conferees.
4. 7th Conference Theme - Considering "Users/people/software", or "computing as it affects the user". It was suggested co-ordinating user groups at the same time, prior or after the conference, as long as they don't become machine house groups.
5. Venue - The two areas being considered are:
Otago University campus
Queenstown.

6. Papers etc - The quality of the Auckland keynote speakers was very good. There were too many academics/machine companies etc, and not enough practical papers. It was considered requesting papers from significant users rather than calling for papers. It was also considered getting each branch to 'arrange' a paper from a particular area, industry or person. If Conference is held in May 1980, N.Z. authors may be able to present the same paper in Australia.
7. Workshops - The paper format and presentation was queried. More time for questions/discussion/workshop type theme. The paper should be published prior to the conference and more discussion held at the conference. It was considered streaming into small groups (20 max) to enable detailed discussion.
8. A further discussion/reading room must be available for discussion etc. by conferees not wanting to go to papers.
9. A large gap between end of sessions and cocktail hour would not be so bad on campus.
10. It was suggested the Presidents dinner be on the first night with a less formal occasion on the second night.
11. Some problems with transport were encountered.
12. Consider inclusion of 'specialists' eg micro-computers, personal computing in the next exhibition and papers.

M.R. Anderson
FORUM CHAIRMAN

FORUM DISCUSSION ON: HUMAN FACTORS IN TERMINAL USEAGE

As the forum was a follow-on from a paper presented during the technical sessions, Peter Cooke introduced the discussion by summarising the papers key points as:

- * Keyboard terminals are the way of the future
- * Manufacturers criteria is different to the eventual end users
- * It is thus the responsibility of the software designer to consider the human factors to help the user through their screen interplay, especially for non-DP users.

Following a diagnosis of some amusing stories relating to the deficiencies of existing systems, the forum reviewed many aspects covering:

1. System must be able to guide the user through problem areas.
2. Assistance maybe either unprompted when input errors occur, or user requested via a 'HELP-BUTTON' concept.
3. Two levels of assistance/prompting may be needed - a fully automated path for a new operator, and a short cut route provided to minimise the frustrations for an experienced operator.
4. To soften the interplay, the system should be more personalised in its responses - (i.e. responding to a personal log-on with a reply 'GOOD MORNING JIM').
5. It is important to tell the user the status of his data - especially when down-time occurs.

- Is there, and where is the malfunction?
- Did the system go down before or after my data was updated?
- When is the system expected to be back up again?

(Maybe, use a mini on the line to produce a 'last-ditch' message).

6. Because written documentation is often misplaced or out-of-date, the systems should be self documenting, using nested HELP procedures to nurse a trainee through his problem.

Overall, it was agreed that the human interface is a weak area of many terminal systems, and that the systems designer must focus more effort onto this area for future applications.

Summary of the N.Z.C.S. forum on the Selection of Computer Managers, held on Saturday August 19th.

Chairman: Bill Williams, Systems & Programs Limited.

Scribe : Jenni Hill, Computronics Limited

25 people from a wide range of installations attended this forum.

The first point to be clarified in this discussion was the type of installation - small, medium or large,
production, technical or strictly commercial.

Depending on their own back grounds, some participants had some clearly defined selection criteria, while others, although holding opinions (their own or their company's) felt they were not covering all points.

One of the main questions discussed was from what background a Computer Manager should come - technical
or management.

It was felt this depended on the type of installation - technical or production.

People normally start in the EDP field at a comparatively young age these days, and technical experience to a high level is generally more readily accessible to a young person than management experience to a comparable level.

The question then arises as to whether or not, having risen to a reasonable level in technical areas, the prospective Computer Manager should then move out to other departments to gain managerial knowledge, and then return to the Data Processing areas later.

Most participants agreed this sounded good in principle, but seldom produced the required benefits.

To gain a reasonable working knowledge of a department, be it Sales, Accounting, Personnel or whatever, at least one year's experience would be needed, preferably two.

Any less than one year was found to be no less than a complete waste of time and effort, both to the prospective Computer Manager, the Managers of other departments, and the company as a whole, in the experience of a couple of those attending.

The question of lines of reporting then arose. The role of the Computer Manager is not clearly defined. Even at installations which have been in operation for many years, some Computer Managers do not enjoy the same Management responsibilities as Sales Managers for instance. In many instances, the Computer Manager still reports to the Financial Manager.

At installations which have an emphasis on development, it was generally agreed that the manager would require a high level of technical expertise.

In other types of installation, persons with technical backgrounds cannot be cut out in the selection process.

But whatever the type of installation, the basic requirements of the Computer Manager remain the same - he (or she) is the Manager of the Information Services Division.

It was agreed that there is a certain aura of mystery surrounding Data Processing personnel. This is particularly noticeable in long established companies where the installation of Data Processing equipment is often regarded as an intrusion by senior members of staff.

If someone from outside is appointed to the Computer Managers position in such a company he is at an immediate disadvantage. Firstly because he is an outsider, and secondly because he does not have experience of the particular company itself. But - does someone from within have the technical expertise to start an installation?

Where a Computer Manager has been appointed from within a Company to a new Computer Installation, it was pointed out,

when technical experience 'in house' is not available, guidance can be obtained particularly from a wide range of Consultants, and indeed similar installations, through contacts at Hardware Users Groups.

But what makes Data Processing differ from other technical occupations? Basically, the sheer rawness of the field.

It is constantly growing as new technologies are introduced, and then improved upon.

In order to cope with these new developments, the Computer Manager of today needs some technical experience - he must also be adaptable. Data Processing is a fast moving industry, and a Manager must be capable of handling assessments for upgrades, and replacements of hardware and staff recruiting. He must also be able to keep pace with the rest of the Company which may also be growing and changing in many areas.

It was suggested that to be fully cognisant of a Company's day to day progress, a Manager, no matter of what department, should spend half of his time out of his own department.

In small or medium installations this may not be possible, because of the number of staff.

In large installations, the Computer Department may be quite separate physically, so such day to day involvement outside one's own area may not be practicable.

So the question is really one of balance.

If the Manager is too technically oriented, he will spend too much, if not all, of his time in Data Processing. For this he is likely to lose the respect of fellow Managers, and depending on the level of his involvement in his own area, may lose the respect of his own staff as well.

On the other hand, if he spends too much time removed from his own area, though he may gain respect from fellow Managers, he will lose respect from his staff because he will rapidly lose touch with the situation - in DP one needs to keep ones finger on the pulse.

The smaller the installation, the harder the Computer Manager's job - he will have to provide technical advice, whereas this is not necessarily the case in a large installation, where people with technical skills will be employed in middle management positions where their technical skills can be used to the full.

If a potential Computer Manager is exposed to other departments for a couple of years, how well does he fit back into the Computer Department? At what level does he re-enter the department? It was felt that to a certain extent, one loses finesse after a break. The technical developments that occur in a two year period may be hard to pick up. And will one want to be returned to the Computer Department. Skills are't lost, but they fade. It was thought that very technical people lose the touch of relating to realities of requirements of users departments.

But - you can't keep a good man down - if someone is the right material, he will end up in the right place.

On the other hand, do DP people actually try to take, or have, the opportunity to gain management skills. Maybe the Data Processing industry is not old enough yet to have developed its own methodology.

A person concerned with the credibility of the Data Processing profession will broaden his knowledge and skills and as a result will draw confidence in himself.

At the same time, the aspiring Data Processing Manager should get away from technicalities in order that he can relate to outside management and company activities.

So are Computer Managers born, trained, or do they just happen?

Past experience shows Data Processing people's loyalties are to their profession rather than their companies - are we really a breed of our own?

NOTE: Time allocated for this forum ran out before the subject had been fully discussed.

Those attending were asked to send in a scaled range of Selection Criteria they believed should be used in the selection of a Computer Manager.

The response to this was sparse and we have been unable to draw any positive conclusions from it.

Summary of the N.Z.C.S. forum on Telecommunications
held on Saturday August 19.

Chairman : Alan Wilkinson, University of Canterbury
Scribe : Alex Sergejew, University of Auckland School
of Medicine.

The Legal Situation

It was felt that perhaps the N.Z.C.S. could help clarify some points, but it was generally believed that :

- i) a computer bureau may send/get data to a customer
- ii) the New Zealand Post Office (NZPO) provides a data carrier service, and charges the bureau with a loading for this
- iii) the owner of the system has the responsibility for ensuring that his equipment is compatible with NZPO standards
- iv) these standards were generally held to be "reasonable" and indeed the NZPO tends to be quite co-operative and pragmatic.
- v) message-switching within a company/organisation was OK, but if some processing is done and then the information is sent to another company, this is acting as a bureau.

The status of automated dialing was unclear, but the existence of such systems was noted.

Standards/Tariffs

NZ is a member of the OECD and therefore has a role in the development of international standards and tariffs, as well as in the local context.

The Communications Commission is the body to approach in this regard, having Government and private representation as well as the NZPO.

The X25 system represents an international standard for a telecommunications protocol, but there appear to be compatibility problems already, with different French, German

and Canadian versions.

Different manufacturers have large investments in their own standards, and not all can accommodate X25, for example.

The PO line should be quite transparent to the data; protocols are a separate problem.

There appears to be a need for tariffs related to the amount of information sent, rather than the current system. Overseas trends are in this direction.

A Post Office Supplied Network ?

There was some interest in something like a PO-supplied packet-switching network, perhaps working-in with the Government networks.

This was seen to have ramifications in terms of the tariff situation, geographic coverage and data security/integrity.

It was suggested that this would enable utilization of recent technological in the telecommunications field, in satellites, fibre optics and digital networks, for example, whilst not re-inventing the wheel.

The NZPO's experiences with the Datel 200 services was alluded-to.

Summary

A sense of urgency regarding the situation should be conveyed to the National Council.

A seminar may be indicated, but the demand should be established first.

The role of the NZCS should not be to attempt to establish specifications itself, but rather to establish an on-going relationship promoting the interplay of information between the parties concerned.

CHOOSING BETWEEN SEQUENTIALLY AND INDEXED-
SEQUENTIALLY ORGANISED FILES FOR SEQUENTIAL
PROCESSING

Author Owen Hanson,
The City University Business School,
LONDON, U.K.

ABSTRACT

The effect of blocking, hit rates, update patterns and the available record formats on the choice between indexed-sequential and sequential files has been analysed. A number of tables, based on a random distribution of updates to these files, have been provided to assist the file designer in making a choice between them, and pointers have also been given to the effect of variations in the conditions analysed.

Key words and phrases: Sequential, indexed sequential, blocking, hit rates, run times block hit rates, block size, record formats, skip sequential processing.

C.R. Categories: 3.70, 3.72, 3.73, 3.74, 3.79 6.34

1. Introduction

Sequential files can be stored on magnetic tape or magnetic disc, and the choice will normally fall on tape. Cases in which disc may be more appropriate include: the need for infrequent direct access; low hit rates, and hence considerable wasted tape passing time; a demand for very high transfer rates, coupled with very short record processing times; small files. These and other cases are considered in more detail in¹.

When a sequential file is stored on magnetic disc, the designer will usually consider whether to organise it sequentially or indexed-sequentially. Indexed files take up more space, use more complex and slower software, and require frequent index references. Despite the precautions that can be taken to ensure indexed-sequential file efficiency², sequential reference to such files will increase run times by between 5% and 33% if every record is to be processed. As the number of blocks hit becomes lower, the position changes, and at some stage there is a break-even point. This depends on a number of factors that are examined below.

Skip-sequential processing of indexed-sequential files will allow records, tracks and sometimes even cylinders (IBM terminology) or seek-areas (ICL) to be skipped. Because an index is available this can usually be arranged without difficulty^{3,4,5}. An equivalent skipping of individual records stored in sequential files is possible only if the key of the record is stored separately from the data, so that there is time for the key to be read in, compared with a desired key, and the data to be read in if required. This type of record format is available in some cases^{3,4}. However, sequential files can never offer the skipping of whole tracks or seek-areas, as each record key has to be examined.

Indexed-sequential files thus provide the possibility of reducing run times in comparison with either type of sequential file (key stored separately, or embedded in the data). The question of choice will be decided by the distribution and rate of hits (i.e. updates) in a given run, the blocking factor used, the number of blocks per track and the available formats for sequentially stored records.

2. Block Hit Rates

When records are stored separately on a magnetic disc, the number of records that have to be processed (whether updated, referenced, deleted etc.) can be calculated directly from the record hit rate. If the hit rate for a file containing 400,000 records is 1%, 4000 records will be processed.

As soon as records are blocked, this simple relationship ceases to hold. If a single record in a block needs to be processed, the whole block has to be read in. The extent to which blocking of records will increase the effective hit rate for the file depends on the distribution of hits through the file. This distribution is often either unknown, or believed to be random; for this reason the random case will be examined first.

a) Random Updates

When a block containing n records has an average random probability p of being accessed, then the probability that any single record in the block will not be accessed is $(1-p/n)$, and hence that no record in the block will be accessed is

$$(1-(p/n))^n \quad \text{I}$$

This is the first term of the Binomial expansion giving the probabilities that 0, 1, 2, ... m records in the block will be accessed. The probability that one or more records will be accessed, and thus that the block must be read in, will therefore be

$$1-(1-(p/n))^n \quad \text{II}$$

The Binomial method of calculating probabilities is accurate (see⁶ for further discussion) but it has the disadvantage that a fresh calculation has to be carried out for each block size. As the number of records in the block increases, the Binomial expressions in I and II tend towards the limits set by the expressions

$$e^{-(p/n)} \quad \text{III}$$

for the probability that no records will be accessed, and hence

$$1 - e^{-(p/n)} \quad \text{IV}$$

for the probability that the block will need to be accessed. Equations III and IV do not depend on n , the number of records per block, but on the ratio (p/n) which is the record hit-rate. They represent the case when n is large, and give a fairly accurate forecast of the proportion of blocks that need to be accessed for values of n greater than 10. A number of theoretical ways of

calculating these values have been reviewed⁶. In practice these do not yield different results from those presented here. Equation II has been used to calculate the probability that any given block will be accessed, for a range of average block hit rates from 0.01 to 8.00 and for block sizes from 1 to 50 records. The figures are given in Table 1. Equation IV has been used to predict the results for an infinite block size; for all practical purposes a block size of fifty records is indistinguishable from 'infinite', and even block sizes of 10 or more records shown maximum errors of the order of five percent. This is the reason that the curve derived from IV has been recommended as a design tool in the author's previous work⁷. The divergence of predictions from this curve for smaller block sizes is shown in Diagram 1. It should be pointed out that the average number of hits per block is represented on a logarithmic scale, and thus that the divergences are greater than at first appears.

An average block hit rate of 1.00 means that only three quarters of two-record blocks need to be read in, and this drops to less than two out of three for blocks containing six or more records. However, when the effect of blocking records with a fixed hit-rate is investigated, the position changes. Diagram 2 shows the percentage of blocks that has to be read as a function of record hit-rate and records per block. The description block activity is given to these figures, to distinguish them from average hits per block.

Diagram 2 presents the results of blocking decisions in such a form that the user can determine block activity for any blocking factor chosen, by drawing a horizontal line through the record hit-rate, and reading off block activities at the intercepts. Taking the example of a record hit-rate of 0.01, or one active record in every hundred, the result of blocking will be as shown in Table 2.

The figures in column two of Table 2 are derived from those in Table 1, and demonstrate the burden in inactive records that have to be read, which is caused by blocking low activity records in large blocks. No attempt has been made to provide a comprehensive table, as any figures required can be taken from Table 1 or calculated using equations II or IV.

It is clear from the data provided that there will be occasions

when the file designer will choose not to use large blocks, and so avoid the high block activity this causes. In some cases the pattern of accesses will be known and the figures derived in Table 2 will not be appropriate. Two of the most common patterns are considered below.

b) Grouped Updates

If most of the records to be accessed are in groups, the objection to large blocks is very much reduced. Most blocks would remain inactive, and the benefits of blocking would make it likely that the file designer would choose as large blocks as possible. When active records can be identified beforehand they can be blocked in groups, and so provide large blocks with the same activity as that of individual records. The usual situation is less clear-cut than this, but is still highly advantageous.

c) Regular Updates

If every fifth, tenth, ...ⁿth record is accessed, the effects of blocking can be precisely calculated. Increases in block activity will tend to be step-wise, and it is often advisable to choose small blocks. However, such a regular pattern is both unusual and, if it does occur, easy to plan for in other ways.

Most update patterns are mixtures of the special cases dealt with above, and the usual solution is to treat them as random. Even if block activity considerations do not limit the size of blocks employed, there is usually competition between files for the available space in main storage, and this is discussed fully in ¹

3. Track and Cylinder Hit Rates

As it is possible, using the track index, to skip whole tracks (and sometimes even cylinders) of an indexed-sequential file, the track and cylinder hit rates may be important. When hits are randomly distributed, the expected number of hits per track or cylinder can be used, as if they were large blocks, to predict the probability that any given track or cylinder can be skipped. It is very unusual for cylinders to be skipped in a random update situation, but this may happen if record references are grouped.

Every block in a sequential file requires checking, in order to ensure that no record in it needs to be processed. For this reason tracks can only be skipped if blocks take up a whole track.

4. Record Formats

Indexed-sequential files on IBM systems require that each record or block should have a separate key field, to allow the record to be checked without the need to read it into main storage. Sequential files do not require this format, as many sequential systems are based on reading every block into main storage, whether they require processing or not. Some systems do allow skip-sequential processing of sequential files, by providing the possibility of using the separate key format^{3,4}.

5. Sequential versus Indexed Files

In comparing the two types of file, the most useful starting point is that of the inevitable additional time an indexed-sequential file will take, when newly created, in comparison with a sequential file.

In sequential retrieval, the cylinder index (and any higher indices) will only be referenced once at the start of the run. Head movement to the first data cylinder will be dependent on the size and position of the file. All further head movements will be from the current cylinder to the next (unless a cylinder can be

skipped), and so will be of minimum duration:- 10-25 milliseconds for modern exchangeable disks. For drums or fixed head disks all that is involved is electronic head-switching, which takes negligible time. The length of a sequential retrieval run will be slower than an equivalent run using a sequentially organised file for the following reasons.

a) References to the cylinder index, and any higher indices

This is not likely to take longer than 100 milliseconds, and average times are much lower, e.g. 30 milliseconds for the IBM 3330, and 60 milliseconds for the 2314. Thus the addition to a given run time will not be significant.

b) Reference to the track index

There will be a rotational delay of half a revolution on average before the required track index entries are detected. This will be repeated for every track on the cylinder, skipped or not, and will thus represent a sizeable addition to run times.

c) Increased file size

The track index will take up part of the first track of each cylinder. Overflow areas, even if empty, will take additional space. This will mean that the data requires more direct access storage space than would apply for a sequential file. Timing will be increased by

(i) Extra cylinder-to-cylinder head movements

(ii) An increased number of data tracks, each of which will require index entries, and so will add to the time required for rotational delays described in b).

As the case of sequential retrieval before additions is being examined, the items given above do not include overflow retrieval. However, there is more housekeeping activity in handling an indexed sequential file than the corresponding sequentially organised file. This will add to CPU usage, and the greater software module size will mean that more space is required in main storage.

To put into perspective the difference in timing between the sequential retrieval of all the records in a sequential file and those same records in an indexed sequential file, general figures have been compared in Table 3. The minimum record read figures are calculated on the assumption that records can be processed as quickly as they are read - i.e. that processing can be completed

while the device is traversing the gap between one record and the next.. This is seldom possible. The more likely situation is that one or more revolutions will be 'lost' between reading successive blocks, because processing of one record or block is not completed until the start of the next has already been passed. Thus the probable record read figures are based on the number of blocks and the time for a full track rotation. The constant multiplier would be 1 or 2 or 3 etc. for individual blocks processed using a single buffer; double buffering would reduce it to $\frac{1}{2}$ or $\frac{2}{2}$ or $\frac{3}{2}$ etc., and larger numbers of buffers would reduce the figures further. (However, it is normally more useful to increase block size than to provide more input buffers than two or three).

Table 4 gives results for the specific example of a file held on an IBM 3330 disk. From these figures it is clear that the main increase in run time is due to seeking and reading track index entries. The file timings are increased by about 47 seconds as a result of being held as an indexed sequential file. This is more than a 70 percent increase in the special case when processing is very rapid. In the more normal case of a double buffered file with more processing the difference in timing is less than 3 percent.

A newly loaded indexed sequential file is thus very little less efficient than a sequential file for sequential retrieval. The effects of additions on these figures is examined in¹.

6. Comparison of Skip Sequential Processing

This paper shows how the break-even point between the two types of file can be determined, in the case for which both files can be skip-sequential processed. When this applies, the indexed sequential file can only benefit by the skipping of whole tracks or cylinders. The probability of being able to skip a full track, given a random update pattern, is shown in Table 5. The figures tabulated apply for blocks containing multiple records, and for multiple records per track. They are only an approximation for small numbers of records per track, and in the specific case of one logical record per track the figures do not apply at all. It is

clear from these figures that either a low hit rate or a small number of blocks per track, or both, will be necessary for sufficient tracks to be skipped to make up the deficit on time caused by the reading of index items during indexed sequential processing.

The indexed sequential file software will always have to locate the first index item on a cylinder, which will require an average of half a revolution. Further items can be scanned at rotation speeds if tracks are skipped, but half a revolution will be required to pick up the next index item after any record has been located. If n tracks have to be referred to on a cylinder, there will be a total loss of time due to index references of

$$\frac{R}{2} (\text{first index item}) + \frac{nR}{2} (\text{further index items})$$

Every skipped track, on the other hand, will represent a saving, as against sequential scanning of the track, of a full revolution. If there are C data tracks per cylinder, then $(C - n)$ tracks are skipped.

As a guide to breakeven points between skip-sequential on indexed sequential files, and skip-sequential on sequential disc files, put the loss and gain of time equal:

$$\begin{aligned} \text{Then } R(C - n) &= R(n + 1)/2 \\ 2C - 2 &= n + 1 \\ n &= (2C - 1)/3 \end{aligned}$$

The most usual values for C are 10, 19 and 20 data tracks per cylinder. These yield break-even points of 37%, 35.1% and 35% of tracks not referred to. This means that, if higher percentages of tracks than these can be skipped, the file designer should recommend an indexed sequential file; otherwise, a sequential file with separate keys will perform better. Table 5 shows the number of tracks that need not be referred to, for a number of combinations of record (or block) hit rate and records (or blocks) per track, and Diagram 3 presents these figures in visual form. These tables are compiled on the assumption that updates occur randomly. If a more definite pattern exists the file designer should use it; the percentage of tracks that must be skipped to break-even does not change, but most non-random patterns increase the number of skipped tracks because updates are more concentrated. This favours the use of indexed sequential files for higher hit-rates than given

here. There is one exception; the very unusual even update distribution will cause a sequential file with separate keys to be more suitable at all hit-rates than an indexed file.

Table 5 does not cover every combination of hit-rate and records per track. Any given combination will show 34.99% or more of tracks skipped if

(Record/Block Hit Rate x Records/Blocks per Track) $\leq$ 1.05
and 36.79% or more if

(Record/Block Hit Rate x Records/Blocks per Track) $\leq$ 1.00

In these calculations hit rate should be expressed as a decimal figure in the range $0 \leq \text{HIT RATE} \leq 1$.

In the case where a sequential file cannot be skip-sequentially processed, the advantage shown by an indexed file is greater than that given here. The limits shown in Diagram 3 are thus rather conservative, and it is worth considering an indexed-sequential file even when the percentage of tracks that can be skipped is well below the figures stated. Experimental work is in progress in this area, and will be presented in due course.

7. Conclusions

The effects of blocking on the processing of indexed-sequential files, and the break-even point between sequential and indexed-sequential files have been analysed, and guidelines for the use of the two file organisation techniques have been presented. These guidelines are lower limits to the use of indexed-sequential files, which may be superior to sequential files over a wider range of values in some circumstances.

References

1. Designing Computer-Based Information Files, by Owen Hanson, Publ. by Pitmans, 1979.
2. Computer Data-Base Organisation, by James Martin, Prentice-Hall, 1975.
3. IBM Manual - Introduction to IBM Direct-Access Storage Devices and Organisation Methods, C20-1649-9.
4. IBM ISAM, VSAM Manuals
5. ICL Manual R201, Part 4.
6. Hit Ratios, by S. J. Waters, The Computer Journal, Vol. 19 No. 1, January 1976, pp 21-24.
7. Basic File Design by Owen Hanson, publ. by IPC Business Press.

TABLE 1

Average block Hit-Rate	BLOCK SIZE (RECORDS)								
	1	2	3	5	10	15	20	50	∞
0.01	0.010	0.010	0.010	0.010	0.010	0.010	0.010	0.010	0.010
0.02	0.020	0.020	0.020	0.020	0.020	0.020	0.020	0.020	0.020
0.03	0.030	0.030	0.030	0.030	0.030	0.030	0.030	0.030	0.030
0.06	0.060	0.059	0.059	0.059	0.058	0.058	0.058	0.058	0.058
0.10	0.100	0.098	0.097	0.096	0.096	0.095	0.095	0.095	0.095
0.20	0.200	0.190	0.187	0.185	0.183	0.182	0.182	0.182	0.181
0.30	0.300	0.278	0.271	0.266	0.263	0.261	0.261	0.260	0.259
0.40	0.400	0.360	0.349	0.341	0.335	0.333	0.332	0.331	0.330
0.50	0.500	0.438	0.421	0.410	0.401	0.399	0.397	0.395	0.393
0.60	0.600	0.510	0.488	0.472	0.461	0.458	0.456	0.453	0.451
0.70	0.700	0.578	0.549	0.530	0.516	0.512	0.510	0.506	0.503
0.80	0.800	0.640	0.606	0.582	0.566	0.561	0.558	0.554	0.551
0.90	0.900	0.698	0.657	0.629	0.611	0.605	0.602	0.597	0.593
1.00	1.000	0.750	0.704	0.672	0.651	0.645	0.642	0.636	0.632
1.10		0.798	0.746	0.711	0.688	0.681	0.677	0.671	0.667
1.20		0.840	0.784	0.746	0.721	0.714	0.710	0.703	0.699
1.30		0.878	0.818	0.778	0.752	0.743	0.739	0.732	0.727
1.40		0.910	0.848	0.807	0.779	0.770	0.766	0.758	0.753
1.50		0.938	0.875	0.832	0.803	0.794	0.790	0.782	0.777
1.60		0.960	0.898	0.855	0.825	0.816	0.811	0.803	0.798
1.70		0.978	0.919	0.875	0.845	0.835	0.831	0.823	0.817
1.80		0.990	0.936	0.893	0.863	0.853	0.848	0.840	0.835
1.90		0.998	0.951	0.908	0.878	0.869	0.864	0.856	0.850
2.00		1.000	0.963	0.922	0.893	0.883	0.878	0.870	0.865
2.25			0.984	0.950	0.922	0.913	0.908	0.900	0.895
2.50			1.000	0.969	0.944	0.935	0.931	0.923	0.918
2.75				0.982	0.960	0.952	0.948	0.941	0.936
3.00				0.989	0.972	0.965	0.961	0.955	0.950
4.00				1.000	0.994	0.990	0.989	0.985	0.982
5.00					0.999	0.998	0.997	0.995	0.993
6.00					1.000	1.000	0.999	0.998	0.997
7.00							1.000	0.999	0.999
8.00								1.000	1.000

The figures tabulated show the probability that any given block will require to be accessed, for the average block hit-rate and number of records per block stated. In processing a large number of blocks they can be used to predict the proportion of blocks that will be accessed.

TABLE 2

Block Size	Block Activity %	Hits per Block (Avg.)
1	1.0	0.01
2	2.0	0.02
3	3.0	0.03
5	4.9	0.05
10	9.6	0.10
15	14.0	0.15
20	18.2	0.20
50	39.5	0.50
100	63.4	1.00
200	86.6	2.00
400	98.2	4.00

TABLE 3

ELEMENTS OF TIMING	TYPE OF FILE	
	SEQUENTIAL	INDEXED SEQUENTIAL
Higher index references	Nil	100 ms
Seek to start	25-60 ms	25-60 ms
Seeking track index entries (Read adds ~ 5%)	Nil	$m'(\ell-1)R/2$
Minimum record read time	$m \ell R^*$	$m' \ell R^*$
Probable record read time	minimum + $C n R$	$C n R$
Cylinder-to-cylinder time	$(m-1)\tau$	$(m'-1)\tau$

EXPLANATION OF SYMBOLS USED

τ = Minimum time, cylinder-to-cylinder

m = cylinders required for the sequential file

m' = cylinders for the indexed file, where $m' = \frac{m \ell}{(\ell-3/2)} +$

n = number of records in file

ℓ = tracks per cylinder

R = rotation time

C = Integer / 2 (see text, for explanation)

* assuming both are in the same format - ie. CKD with key separate.

+ the figure of 3/2 represents one track for embedded overflow, and half a track for the track index.

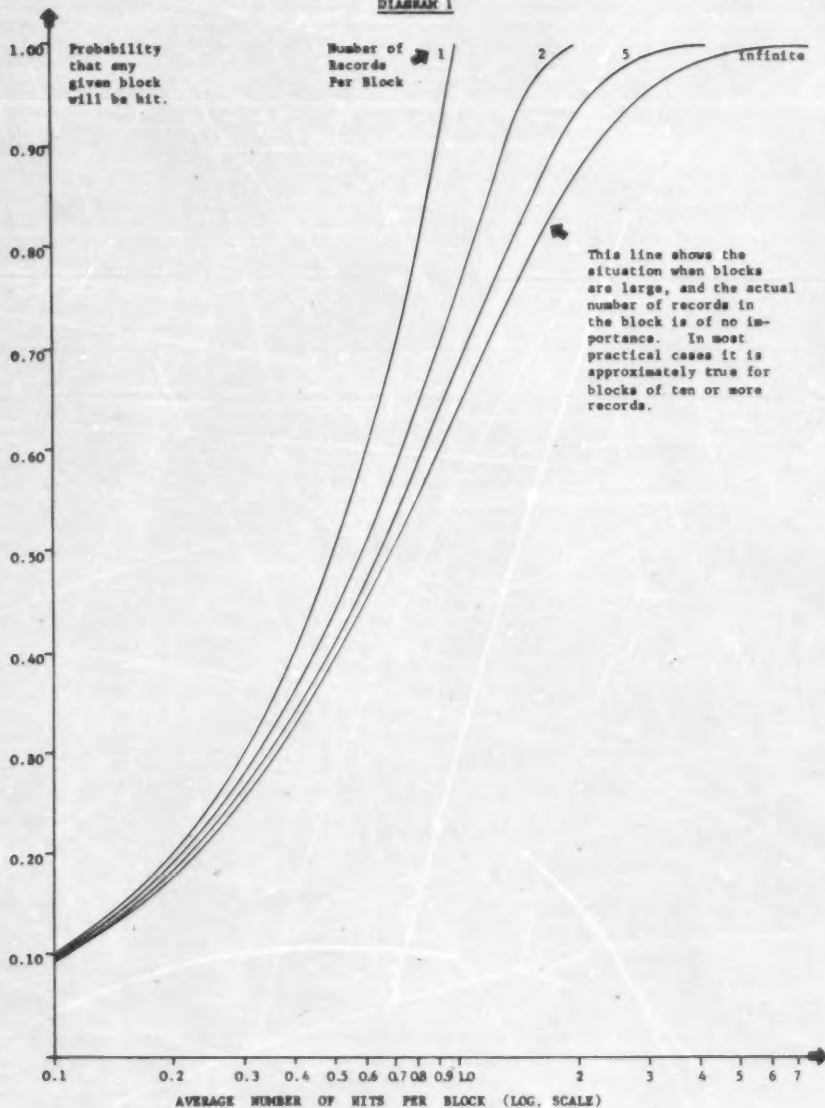
TABLE 4

400,00 Records of 80 Byte Length, 6 Byte Keys, Blocked in Fives.		
Elements of Timing	Type of File	
	Sequential	Indexed Sequential
Higher Index Reference	-	60 mS
Seek to Start of File	30 mS	30 mS
Seeking Track Index Entries	-	38.623 secs
Minimum Record Read Time	63.627 secs	70.725 secs
Probable Record Read Times	63.627 + n x 3340 secs	70.725 + n x 3340 secs
Cylinder to Cylinder Time	2.010 secs	2.650 secs
Total Times	65.7 + 6680C seconds	112.1 + 6680C seconds

TABLE 5
FIGURES TABULATED SHOW THE PERCENTAGE OF TRACKS WITH NO UPDATES

Track Capacity (Records)	Block or Record Hit Rate (%)																			
	1	2	3	4	5	6	8	10	12	15	18	21	25	30	35	40	45	50	60	70
1	99.00	98.02	97.04	96.08	95.12	94.18	92.31	90.48	88.69	86.10	83.53	81.06	77.88	74.08	70.47	67.03	63.76	60.65	54.88	49.66
2	98.02	96.08	94.18	92.31	90.48	88.69	85.21	81.87	78.66	74.08	69.77	65.70	60.65	54.88	49.66	44.93	40.66	36.79	31.12	24.66
3	97.04	94.18	91.39	88.69	86.10	83.53	78.66	74.08	69.77	63.76	58.27	53.26	47.24	40.66	34.99	31.12	25.92	22.31	16.53	12.25
4	96.08	92.31	88.69	85.21	81.87	78.66	72.61	67.03	61.88	54.88	48.68	43.17	36.79	30.12	24.65	20.19	16.53	13.53	9.07	6.08
5	95.12	90.48	86.10	81.87	77.88	74.08	67.03	60.65	54.88	47.24	40.66	34.99	28.65	22.31	17.38	13.53	10.54	8.21	4.98	3.02
6	94.18	88.69	83.53	78.66	74.08	69.77	61.88	54.88	48.68	40.66	33.96	28.37	22.31	16.53	12.25	9.07	6.72	4.98	2.73	1.50
7	93.24	86.94	81.06	75.58	70.47	65.70	57.12	49.66	43.17	34.99	28.37	22.99	17.38	12.25	8.63	6.08	4.29	3.02	1.50	
8	92.31	85.21	78.66	72.61	67.03	61.88	52.73	44.93	38.29	30.12	23.69	18.64	13.53	9.07	6.08	4.08	2.73	1.83		
9	91.39	83.53	76.34	69.77	63.76	58.27	48.68	40.66	33.96	25.92	19.79	13.81	10.54	6.72	4.29	2.73	1.74	1.11		
10	90.48	81.87	74.08	67.03	60.65	54.88	44.93	36.79	30.12	22.31	16.53	12.25	8.21	4.98	3.02	1.83	1.11			
11	89.58	80.25	71.89	64.40	57.69	51.69	41.48	33.29	26.71	19.20	13.81	9.93	6.39	3.69	2.13	1.23				
12	88.69	78.66	69.77	61.88	54.88	48.68	38.29	30.12	23.69	16.53	11.53	8.05	4.97	2.73	1.50					
13	87.81	77.11	67.71	59.45	52.20	45.84	35.35	27.25	21.01	14.23	9.63	6.52	3.88	2.02	1.06					
14	86.94	75.58	65.70	57.12	49.66	43.17	32.63	24.66	18.64	12.25	8.05	5.29	3.02	1.50						
15	86.07	74.08	63.76	54.88	47.24	40.66	30.12	22.31	16.53	10.54	6.72	4.08	2.35	1.11						
16	85.21	72.61	61.88	52.73	44.93	38.29	27.80	20.19	14.66	9.07	5.61	3.47	1.83							
18	83.53	69.77	58.27	48.68	40.66	33.96	23.69	16.53	11.53	6.72	3.92	2.28	1.11							
21	81.06	65.70	53.26	43.17	34.99	28.37	18.64	12.25	8.05	4.08	2.28	1.22								
25	77.88	60.65	47.24	36.79	28.65	22.31	13.53	8.21	4.97	2.35	1.11									
30	74.08	54.88	40.66	30.12	22.31	16.53	9.07	4.98	2.73	1.11										
35	70.47	49.66	34.99	24.65	17.38	12.25	6.08	3.02	1.50											
40	67.03	44.93	31.12	20.19	13.53	9.07	4.08	1.83												
45	63.76	40.66	25.92	16.53	10.54	6.72	2.73	1.11												
50	60.65	36.79	22.31	13.53	8.21	4.98	1.83													
60	54.88	31.12	16.53	9.07	4.98	2.73														
70	49.66	24.66	12.25	6.08	3.02	1.50														
80	44.93	20.19	9.07	4.08	1.83															
90	40.66	16.53	6.72	2.73	1.11															
100	36.79	13.53	4.98	1.83																

DIAGRAM 1



153
 DIAGRAM 2

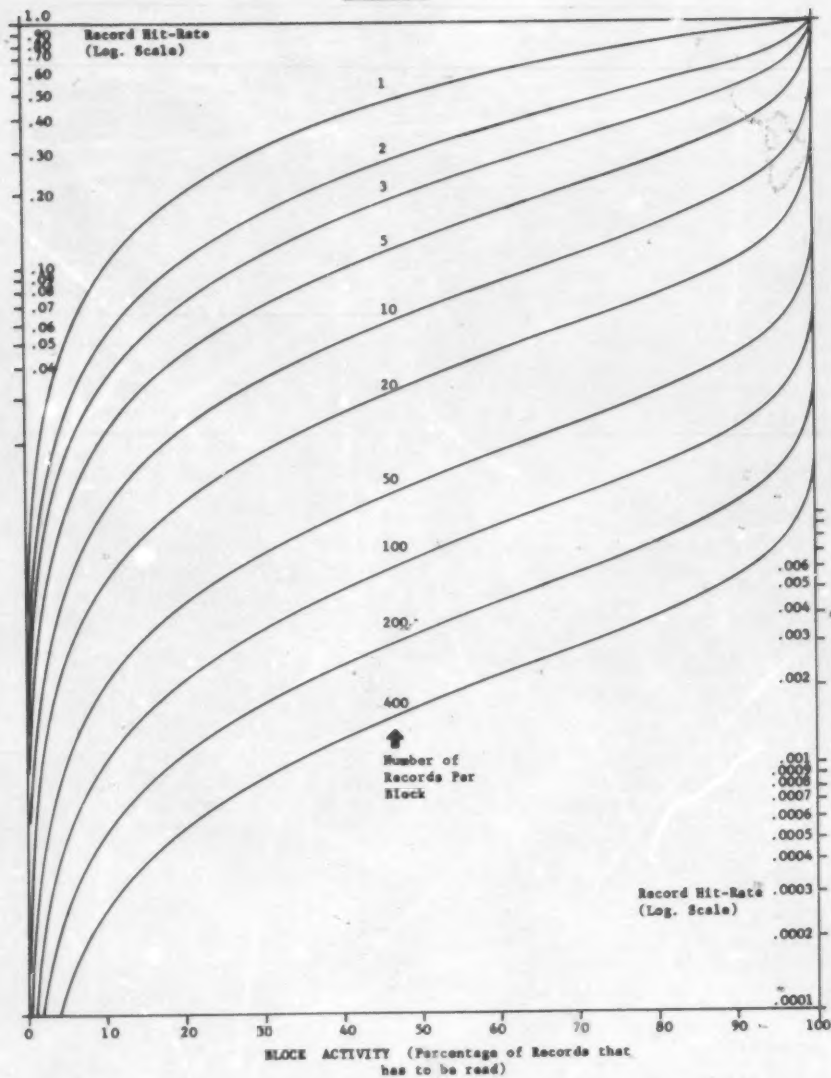
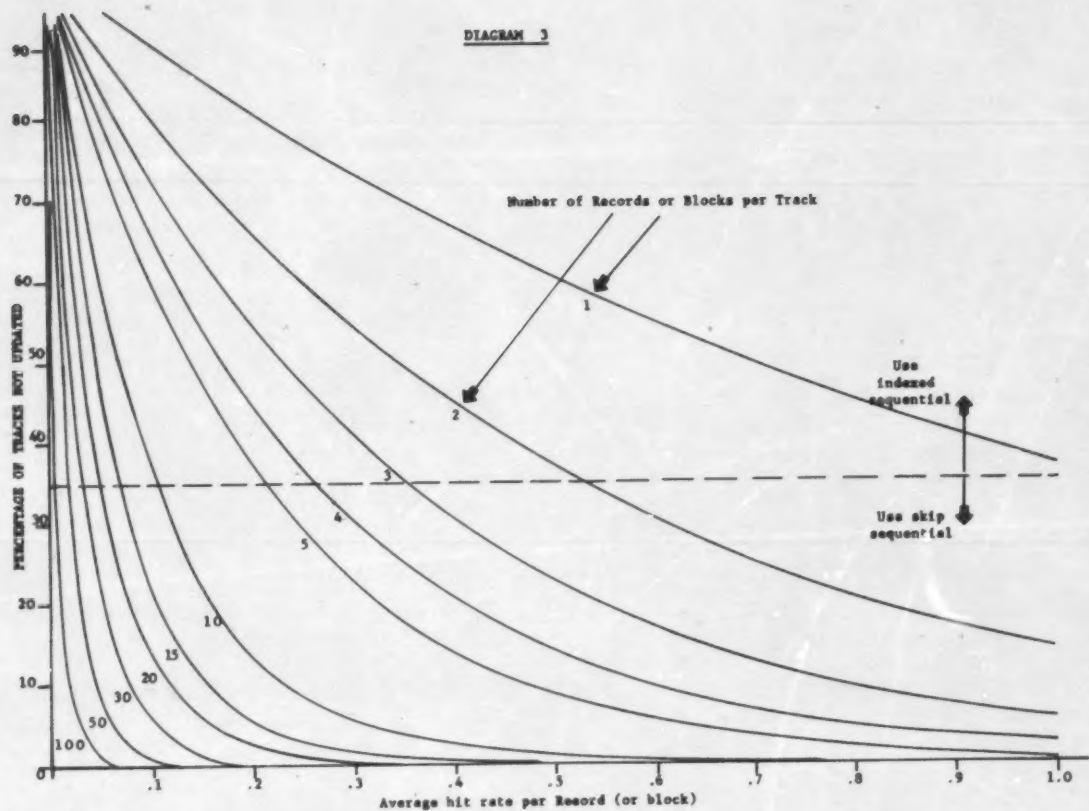


DIAGRAM 3



1.54

TABLE 5
FIGURES TABULATED SHOW THE PERCENTAGE OF TRACKS WITH NO UPDATES

Track Capacity (Records)	Block or Record Hit Rate (%)																							
	1	2	3	4	5	6	8	10	12	15	18	21	25	30	35	40	45	50	60	70	80	90	100	
1	99.00	98.02	97.04	96.08	95.12	94.18	92.31	90.48	88.69	86.10	83.53	81.06	77.88	74.08	70.47	67.03	63.76	60.65	54.88	49.66	44.93	40.66	36.79	
2	98.02	96.08	94.18	92.31	90.48	88.69	85.21	81.87	78.66	74.08	69.77	65.70	60.65	54.88	49.66	44.93	40.66	36.79	31.12	24.66	20.19	16.53	13.53	
3	97.04	94.18	91.39	88.69	86.10	82.53	78.66	74.08	69.77	63.76	58.27	53.26	47.24	40.66	34.99	31.12	25.92	22.31	16.53	12.25	9.07	6.72	4.98	
4	96.08	92.31	88.69	85.21	81.87	78.66	72.61	67.03	61.88	54.88	48.68	43.17	36.79	30.12	24.65	20.19	16.53	13.53	9.07	6.08	4.08	2.73	1.83	
5	95.12	90.48	86.10	81.87	77.88	74.08	67.03	60.65	54.88	47.24	40.66	34.99	28.65	22.31	17.38	13.53	10.54	8.21	4.98	3.02	1.83	1.11		
6	94.18	88.69	83.53	78.66	74.08	69.77	61.88	54.88	48.68	40.66	33.96	28.37	22.31	16.53	12.25	9.07	6.72	4.98	2.73	1.50				
7	93.24	86.94	81.06	75.58	70.47	65.70	57.12	49.66	43.17	34.99	28.37	22.99	17.38	12.25	8.63	6.08	4.29	3.02	1.50					
8	92.31	85.21	78.66	72.61	67.03	61.88	52.73	44.93	38.29	30.12	23.69	18.64	13.53	9.07	6.08	4.08	2.73	1.83						
9	91.39	83.53	76.34	69.77	63.76	58.27	48.68	40.66	33.96	25.92	19.79	13.81	10.54	6.72	4.29	2.73	1.74	1.11						
10	90.48	81.87	74.08	67.03	60.65	54.88	44.93	36.79	30.12	22.31	16.53	12.25	8.21	4.98	3.02	1.83	1.11							
11	89.58	80.25	71.89	64.40	57.69	51.69	41.48	33.29	26.71	19.20	13.81	9.93	6.39	3.69	2.13	1.23								
12	88.69	78.66	69.77	61.88	54.88	48.68	38.29	30.12	23.69	16.53	11.53	8.05	4.97	2.73	1.50									
13	87.81	77.11	67.71	59.45	52.20	45.84	35.35	27.25	21.01	14.23	9.63	6.52	3.88	2.02	1.06									
14	86.94	75.38	65.70	57.12	49.66	43.17	32.63	24.66	18.64	12.25	8.05	5.29	3.02	1.50										
15	86.07	74.08	63.76	54.88	47.24	40.66	30.12	22.31	16.53	10.54	6.72	4.08	2.35	1.11										
16	85.21	72.61	61.88	52.73	44.93	38.29	27.80	20.19	14.66	9.07	5.61	3.47	1.83											
18	83.53	69.77	58.27	48.68	40.66	33.96	23.69	16.53	11.53	6.72	3.92	2.28	1.11											
21	81.06	65.70	53.26	43.17	34.99	28.37	18.64	12.25	8.05	4.08	2.28	1.22												
25	77.88	60.65	47.24	36.79	28.65	22.31	13.53	8.21	4.97	2.35	1.11													
30	74.08	54.88	40.66	30.12	22.31	16.53	9.07	4.98	2.73	1.11														
35	70.47	49.66	34.99	24.65	17.38	12.25	6.08	3.02	1.50															
40	67.03	44.93	31.12	20.19	13.53	9.07	4.08	1.83																
45	63.76	40.66	25.92	16.53	10.54	6.72	2.73	1.11																
50	60.65	36.79	22.31	13.53	8.21	4.98	1.83																	
60	54.88	31.12	16.53	9.07	4.98	2.73																		
70	49.66	24.66	12.25	6.08	3.02	1.50																		
80	44.93	20.19	9.07	4.08	1.83																			
90	40.66	16.53	6.72	2.73	1.11																			
100	36.79	13.53	4.98	1.83																				